UNIVERSIDADE FEDERAL DE SANTA CATARINA
DEPARTAMENTO DE AUTOMAÇÃO E SISTEMAS

Daniel Martins Lima

PREDICTOR-BASED ROBUST CONTROL OF DEAD-TIME PROCESSES

Florianópolis

2015

Daniel Martins Lima

PREDICTOR-BASED ROBUST CONTROL OF DEAD-TIME PROCESSES

Tese de doutorado submetida ao Programa de Pós-Graduação em Engenharia de Automação e Sistemas para a obtenção do Grau de Doutor em Engenharia de Automação e Sistemas.
Orientador: Prof. Dr. Julio Elias Normey-Rico
Coorientador: Prof. Dr. Tito Luís Maia Santos

Florianópolis

2015

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Lima, Daniel Martins
PREDICTOR-BASED ROBUST CONTROL OF DEAD-TIME PROCESSES / Daniel Martins Lima ; orientador, Julio Elias Normey-Rico ; coorientador, Tito Luís Maia Santos. - Florianópolis, SC, 2015.
212 p.

Tese (doutorado) - Universidade Federal de Santa Catarina, Centro Tecnológico. Programa de Pós-Graduação em Engenharia de Automação e Sistemas.

Inclui referências

1. Engenharia de Automação e Sistemas. 2. Processos não-lineares. 3. Compensação de Atraso de Transporte. 4. Controle Robusto. 5. Preditor não-linear. I. Normey-Rico, Julio Elias . II. Santos, Tito Luís Maia . III. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Engenharia de Automação e Sistemas. IV. Título.

Daniel Martins Lima

PREDICTOR-BASED ROBUST CONTROL OF DEAD-TIME PROCESSES

Esta tese de doutorado foi julgada aprovada para a obtenção do Título de **"Doutor em Engenharia de Automação e Sistemas"**, e aprovada em sua forma final pelo **Programa de Pós-Graduação em Engenharia de Automação e Sistemas.**

Florianópolis, 20 de Maio de 2015.

Prof. Dr. Rômulo Silva de Oliveira
Coordenador do Curso

Banca Examinadora:

Julio Elias Normey-Rico – DAS – UFSC
Presidente – Orientador

Alexandre Trofino Neto – DAS – UFSC

Daniel Juan Pagano – DAS – UFSC

Darci Odloak – Eng. Química – USP

Ricardo H. C. Takahashi – Dep. Matemática – UFMG

Rodolfo C. C. Flesch – DAS – UFSC

À minha família que me permitiu chegar até aqui.

AGRADECIMENTOS

À minha família pelo eterno apoio, especialmente ao Jedi pelos momentos de descontração.

À Nathália pela companhia nestes últimos quatro anos.

Aos amigos do laboratório pela companhia nos últimos 2 anos.

Aos meus orientadores Julio e Tito, que com seus direcionamentos me permitiram terminar esta tese em um tempo fora do comum.

Aos professores do DAS, que cuidaram de toda minha formação acadêmica.

À Petrobras e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pelo auxílio financeiro.

A great step for a man, a teeny-tiny step for mankind.

Paraphrase of PHD Comics - Strip "Giant Leap"

RESUMO ESTENDIDO

Esta tese trata do problema de controle robusto de sistemas não-lineares com atraso utilizando estruturas de compensação de atraso.
Como já descrito na literatura, três são os problemas causados pela presença de atraso de transporte: (i) os efeitos das perturbações não são notados até se passar o tempo do atraso, (ii) o efeito da ação de controle demora para ser notado na variável controlada, e (iii) a ação de controle que é aplicada no instante atual tenta corrigir uma situação que se originou tempos atrás.
Uma das mais utilizadas soluções para evitar (ou atenuar) esses efeitos é o uso do Preditor de Smith (SP - *Smith Predictor*). Preditores são estruturas que permitem o controle de processos com atraso a partir de um modelo sem atraso, o que simplifica o ajuste do controlador. Uma importante propriedade do Preditor de Smith vem do fato de que a robustez do sistema de malha fechada resultante não depende do valor nominal do atraso. Esta propriedade, no entanto, não é válida para qualquer preditor. Por exemplo, algoritmos de controle preditivo (MPC - *Model Based Predictive Controllers*) definem implicitamente estruturas preditoras, mas, como já foi mostrado na literatura, no caso específico do GPC (*Generalized Predictive Control*), o preditor ótimo definido implicitamente faz com que a robustez do sistema dependa do valor nominal do atraso. Também já havia sido mostrado que, substituindo este preditor implícito por um Preditor de Smith Filtrado (FSP - *Filtered Smith Predictor*), resulta em um controlador mais robusto que herda as características do SP.
Assim, os objetivos desta tese são: (i) Estudo do algoritmo preditivo *Dynamic Matrix Control* (DMC), através de uma estrutura FSP, e propor modificações que permitam melhorar a rejeição de perturbações e/ou aumentar a robustez do sistema; (ii) análise e implementação de uma estrutura baseada no FSP para sistemas não-lineares.
Os algoritmos de controle preditivo, ou MPC, emergiram durante as últimas três décadas como uma poderosa solução de controle, e obtiveram um impacto significativo na indústria, como já mostrado em diversos trabalhos. No entanto, apesar de grandes avanços teóricos e do fato de que os processos industriais são, em geral, não lineares, a maioria das técnicas de controle aplicadas na indústria são baseadas em modelos lineares. Algoritmos MPC simples baseados em modelos de resposta ao degrau (ou impulsiva) sem garantia de estabilidade são

os mais comuns na indústria, principalmente em refinarias e plantas petroquímicas. Algumas razões para isso são: (i) os processos possuem comportamento estável em malha aberta e ajustando adequadamente os parâmetros do controlador é possível obter a estabilidade do sistema em malha fechada, e (ii) modelos lineares são suficientes quando o processo está operando próximo de um ponto de operação. Desta forma, a análise das propriedades de malha fechada desses controladores, como velocidade de rejeição de perturbação e robustez, é muito importante para a indústria de processos, já que é possível obter modificações simples e úteis que melhoram o desempenho de aplicações reais.

Assim, neste trabalho, o algoritmo preditivo DMC será interpretado através da estrutura FSP de forma que os efeitos do atraso no sistema de malha fechada possam ser entendidos. Esta abordagem foi escolhida por permitir que várias técnicas de sintonia já desenvolvidas para o FSP possam ser aplicadas ao DMC. Será mostrado que o algoritmo DMC precisa apenas de pequenas modificações para adquirir as vantagens fornecidas pela estrutura FSP.

O segundo tópico deste trabalho trata de estruturas preditoras para sistemas não-lineares. Seguindo as ideias propostas para o caso linear, neste trabalho será proposto o Preditor de Smith Filtrado para Sistemas Não-Lineares (NLFSP - *Nonlinear Filtered Smith Predictor*), que permitirá melhorar as características de robustez e rejeição de perturbação de sistemas não lineares. Já há trabalhos evidenciando algumas vantagens do FSP para sistemas não-lineares, no entanto não há provas nem uma análise formal de suas propriedades.

O FSP linear possui as seguintes características: (i) a resposta nominal para mudanças de referência não é afetada pela inserção do filtro de predição; (ii) a robustez pode ser melhorada ajustando o filtro adequadamente; (iii) o filtro de predição pode ser ajustado para acelerar a rejeição de perturbações. Vários exemplos de simulação são apresentados no documento para ilustrar os resultados teóricos apresentados. Em particular, se aplicam os resultados a processos da indústria do petróleo e petroquímica onde os controladores preditivos têm um grande impacto.

Palavras-chave: Processos não-lineares. Compensação de Atraso de Transporte. Controle Robusto. Preditor não-linear.

ABSTRACT

This thesis deals with the analysis and design of predictor-based robust controllers for processes with dead time. The main objectives are: (i) to analyze the effect of the predictor structure in the closed-loop behaviour and robustness of linear and nonlinear controllers; (ii) to propose better predictor structures to improve robustness and performance of control loops; (iii) to apply the results in simulated and real industrial processes, mainly for the petroleum industry. The results of this thesis are: an improvement on the well-known Dynamic Matrix Control (DMC) algorithm, from the Model Predictive Control (MPC) family, and a predictor for nonlinear systems with time delay based on the Smith Predictor. Concerning the MPC, in this work, an improved industrial MPC controller based on the widely used DMC approach is presented. A MIMO filter is included in the prediction model of the controller in order to achieve two important advantages when compared to traditional industrial DMC: (i) disturbance rejection response can be speeded up and (ii) robustness can be improved, mainly when errors in the estimation of the delays are considered. The filter properties are demonstrated by means of an equivalent analysis of the unconstrained DMC using a dead time compensation (DTC) approach, namely the Filtered Smith Predictor. Moreover implementation and tuning of the filter is simple and intuitive. Simulation results using a water-methanol distillation column are presented to illustrate the advantages of the proposed approach. For the case of nonlinear processes with time delay, a Nonlinear Filtered Smith Predictor (NLFSP) structure is proposed for nonlinear systems. It will be shown that the NLFSP maintains the characteristics of the linear Smith Predictor and that, with appropriate tuning, it can increase the robustness of the closed-loop system. The NLFSP is applied to various examples and case studies to demonstrate these characteristics.

Keywords: Nonlinear Process. Dead Time Compensation. Model Predictive Control. Robust Control. Nonlinear Predictor.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

2DOF	Two Degree of Freedom
AR	Auto Regressive
CARIMA	Controller Auto-Regressive Integrated Moving-Average
DTC	Dead Time Compensation
DMC	Dynamic Matrix Control
FDMC	Filtered Dynamic Matrix Control
FSP	Filtered Smith Predictor
FOPDT	First Order Plus Dead Time
GPC	Generalized Predictive Control
ISS	Input to State Stability
MFD	Matrix Fraction Description
MIMO	Multiple-Inputs Multiple-Output
MPC	Model Predictive Control
NLFSP	NonLinear Filtered Smith Predictor
NMPC	Nonlinear Model Predictive Control
PNMPC	Practical Nonlinear Model Predictive Control
SISO	Single-Intput Single-Output
SP	Smith Predictor
SQP	Sequential Quadratic Programming

LIST OF SYMBOLS

t Represents the discrete time instant

q The delay operator, i.e., $y(t)q^{-1} = y(t-1)$

z Is a complex variable that represents the frequency domain used with the $\mathcal{Z}$-Transform

Δ Represents the first difference operator, which can be defined using q ($\Delta = 1 - q^{-1}$), or z ($\Delta = 1 - z^{-1}$)

$|\cdot|$ Defines the Euclidian norm, also known as Norm-2

x Variables in lowercase are scalars

$\mathbf{x}$ Variables in lowercase and bold are vectors

$\mathbf{X}$ Variables in uppercase and bold are matrices

$\mathbb{X}$ Represents sets

$X(z)$ Defines a transfer function

$\mathbf{X}(z)$ Defines a matrix transfer function

$\mathbf{I}$ Defines the identity matrix

$\mathbf{1}$ A vector whose elements are all ones, or, depending on the context, a block diagonal matrix whose diagonal elements are vectors with all elements equal to one

$\mathbf{0}$ A vector whose elements are all zeros

$\mathbf{a}_{[i,j]}$ Represents a signal sequence: $\mathbf{a}_{[i,j]} \triangleq \{\mathbf{a}(i), \mathbf{a}(i+1), \ldots, \mathbf{a}(j)\}$

$\mathbf{0}_{[i,j]}$ A suitable sequence taking the null value

T_s Sampling period

N_1 Initial prediction horizon

N_2 Final prediction horizon

N_i Size of the prediction horizon, which is defined as $N = N_2 - N_1 + 1$

N_u Control horizon

$\mathbf{Q_y}$ Weights of the future errors used in MPC

$\mathbf{Q_u}$ Weights of the future control increments used in MPC

$\kappa(.)$ Represents a generic controller that can be linear or nonlinear

SUMMARY

1 INTRODUCTION

As pointed out by Palmor in [4] three are the main difficulties introduced by the delay: (i) effects of the disturbances are not noticed until the dead time has elapsed, (ii) the effect of the control action takes some time to be noticed in the controlled variable, and (iii) the control action that is applied based on the actual error tries to correct a situation that originated some time before.

Since the seminal work of Smith [5], one solution to avoid (or attenuate) these effects is the use of the Smith Predictor (SP). Predictors are structures which compensate the dead time of the process which simplifies the control design because the controller can be designed for the delay-free nominal model [4, 6]. The predictor structure is independent of the controller, however, for a fixed controller, the closed-loop properties (robustness, disturbance rejection, etc.) are different for each predictor scheme because each one computes the prediction differently.

An important property of the Smith Predictor comes from the fact that robustness margins are not related to the nominal dead-time value. This characteristic is very interesting since it is not necessary to consider nominal dead time length from a robust stability point of view. This property, however, does not hold for any predictor. Take, for example, the case of the Optimal predictor, which is used in many Model Predictive Control (MPC) algorithms, and the Generalized Predictor, proposed in [7]. Their robustness properties, as demonstrated in [3] and [8], respectively, are dependent on the nominal dead time, i.e., for different values of the dead time, they exhibit different robustness behaviour.

The subject of predictors is specially important in the MPC context. However, few are the works that consider the effect of the predictor on the closed-loop properties of the system. MPC has emerged as a powerful practical control technique during the last three decades [2]. It is one of the few advanced control techniques that have had a significant impact on industrial process control and many works have reported the advantages of its use in different plants [9].

Also, the MPC academic community has been very active in the last years, approaching different aspects of MPC algorithms. Efforts have been done to obtain conditions for stability guarantee in linear MPC [10, 11] or to extend the qualities of linear algorithms to nonlinear MPC [12] or hybrid MPC. MPC presents a series of advantages over other methods, amongst which the following stand out [3]:

- it can be used to control a great variety of processes, ranging from those with relatively simple dynamics to more complex ones, including systems with long dead time, nonminimum phase and unstable ones;

- the multivariable case can easily be dealt with;

- it intrinsically compensates for dead times;

- it introduces feedforward control in a natural way to compensate for measurable disturbances;

- its extension to the treatment of constraints is conceptually simple and these can be systematically included during the design process.

The dead-time compensation is of special interest. MPC algorithms have, basically, two stages, the prediction stage and the control computation stage. The former uses a model of the plant to predict the future behaviour of the process and the latter uses this information to compute the control action based on the optimization of a cost function that takes into account the predicted future output and input. Hence, these algorithms compensate the dead time of the system and thus have predictor structures. However, the predictor is implicit, i.e., the algorithm is implemented in a way that the prediction and the computation of the control action are done at the same time. Nonetheless, for analysis, these steps can be separated as shown in [3]. The implicit nature of the predictor is the main reason why few works in the literature analyzes the impact of the predictor structure on the closed-loop properties in the MPC context.

In [3] it was demonstrated that, for the specific case of the Generalized Predictive Control (GPC), the implicit optimal predictor makes the stability margins of the controller dependent on the nominal dead time. This kind of problem is related to the implicit disturbance model and observer. Also in [3], it was proved that substituting the implicit predictor by an explicit SP-based predictor, called Filtered Smith Predictor (FSP), resulted in a more robust controller that inherited the advantageous characteristics of the SP.

One of the objectives of this thesis is to analyse the Dynamic Matrix Control (DMC) algorithm, described in [13], as was done with the GPC in [3], and verify if it is possible to use the FSP to improve the closed-loop characteristics of this particular controller. Despite its

limitations, e.g., has slow disturbance rejection, can only deal with stable systems, etc. [14], and in spite of the new theoretical developments in MPC, the DMC is still one of the most used MPC controllers in industry.

Despite the fact that industrial processes are nonlinear, most control design techniques in industrial applications are based on linear models. Simple MPCs based on step or impulse response models without stability guarantee are the most common in industry, mainly in refineries and petrochemical plants, where MPC is widely used. The main reasons for this are: (i) processes have a stable behavior in open loop and using long horizons and adequate weighting factors allows for stable closed-loop systems [15], and (ii) linear models provide good results when the plant is operating in the neighborhood of the operating point. In the process industry, the objective is to keep the process around the stationary state rather than perform frequent changes in the operating point, thus a suitable linear model is sufficient [2]. Hence, the analysis of some closed-loop properties of these controllers, like disturbance rejection and robustness, is very important as they can be transformed in simple and useful modifications of the control algorithms to achieve better performance in real applications [3]. Particularly, in multivariable industrial processes, dead times are always used to model the plant behavior. In such models, each signal path between the inputs and outputs may show a different delay. Although these MPC formulations can include multiple delays in a straightforward manner, robustness and/or disturbance rejection can be poor because the controller does not have enough degrees of freedom to achieve a satisfactory trade-off between these two important specifications [3].

Thus, in this work, the DMC is interpreted through a Dead Time Compensator (DTC) structure to understand the effects of dead time on the closed-loop system. This approach uses a transfer function representation for analysis instead of state-space, as was presented in numerous papers, e.g., [14, 16, 17]. This approach was chosen in order to use some specific DTC robust tuning methodologies already developed for MIMO dead time compensators (MIMO-DTC) [1] for improving the robustness and disturbance rejection performance of DMC. Moreover, it is shown that the original algorithm only needs some minor modifications and that the implementation and tuning of this strategy are simple and straightforward, contrary to the solutions based on state-space approach.

1.1 MPC FOR NONLINEAR SYSTEMS

As stated before, in practice processes are nonlinear, but in many cases linear MPC algorithms are sufficient to provide a good closed-loop behaviour. However, there are some instances where this is not true, e.g., when the process dynamics is highly nonlinear or when it is necessary to change frequently the operating point of the process. In these cases, the linear algorithms are not satisfactory, thus, MPC techniques where a nonlinear model of the process is considered were developed, and they now compose the Nonlinear MPC (NMPC) family.

NMPC has received considerable attention from the academic community, which is reflected in the number of books on this subject [18–20] and also in the existence of the triannual IFAC Conference on Nonlinear Model Predictive Control, already in its fifth edition, which brings dozens of papers about this subject.

However, as in the linear case, there are few works that study the role of the implicit predictor of the NMPC algorithms on the closed-loop properties of the system. In [6, 21], the authors analyse the robustness properties of the optimal predictor (the one used in many MPC algorithms), and they prove that its robustness characteristics are dependent on the nominal dead-time, similarly to the linear case. In [22, 23], the authors propose a modification of the optimal predictor to make it more robust, however, it is still dependent on the nominal dead time. Therefore, in this thesis the study of nonlinear predictor schemes is proposed.

1.2 PREDICTORS FOR NONLINEAR SYSTEMS

Predictors for nonlinear systems are discussed in various works. In [24–26], the authors propose a predictor-based controller to stabilize the process with input time-delay, where the delay can be time-varying and state-dependent. However, the proposed predictor structure does not guarantee an offset-free prediction for constant disturbances, which compromises the reference tracking capabilities of the closed-loop system, and it is formulated in continuous time and no comments on the implementation of the predictor are made. In [27], the SP is applied to a class of nonlinear systems, but the analysis of the predictor is done in conjunction with a Globally Linearizing Control (GLC) strategy, which linearizes the process, thus allowing the use of mathematical tools for linear systems. In [28], the authors follow the idea presented in [27],

and extend the SP to a more general nonlinear system model, but, because of how the SP handles disturbances, this structure was dropped in favor of a Kalman filter based predictor, however, the effects of the tuning parameters on robustness are not thoroughly analysed. In [29], a nonlinear system is controlled remotely through a network, which causes a time-delay on the process response. Hence, they propose a SP to compensate the dead time. However, the system is locally controlled by a linearizing compensator, hence a linear SP is used. In [30], the SP is used to control a pure integrative process with dead time and input saturation. Tuning guidelines are suggested to provide good robustness performance in the presence of delay uncertainties. However, no generalization for more general nonlinear systems is provided.

There are also some applications of the FSP for nonlinear system with MPC algorithms. One of the first examples is presented in [3], and other applications can be found in [31–33], in these works a NMPC is used to control the process, which exhibits dead time, and the FSP is used to increase the closed-loop robustness, but no theoretical analysis is made.

Hence, following the ideas presented in [3] for the linear case, a Nonlinear Filtered Smith Predictor (NLFSP) will be proposed in this work to improve the robustness of dead-time nonlinear closed-loop systems. The linear FSP has the following characteristics [34]:

a) nominal set-point performance is not affected by the prediction filter;

b) robustness can be improved by a suitable tuning of the predictor filter;

c) the filter can be used to improve disturbance rejection;

d) it can also be applied to unstable processes.

For the general stable nonlinear system description, only properties (a) and (b) will be proven in this thesis. However, for a particular case of nonlinear systems, those whose nonlinearities lie in the input, all properties will be proven.

1.3 OBJECTIVES AND ORGANIZATION OF THE THESIS

The objectives of this thesis are twofold:

1. Study of the commonly used Dynamic Matrix Control (DMC), which is a linear Model Predictive Control (MPC) algorithm, through a Smith Predictor perspective and the proposition of a modification which will allow disturbance rejection improvement and/or increased robustness.

2. Theoretical analysis with proofs of the FSP structure for nonlinear systems in discrete time, which makes the FSP structure ready to implement in practice. Also, it is important to note that the FSP can be used with any controller, however, it is specially interesting in the MPC context.

This thesis is organized as follows. Chapters 2 and 3 focus on the literature review. In Chapter 2 the DTC structure known as Smith Predictor will be introduced and also its modification, the FSP, which provides an additional tuning parameter that can be adjusted to improve robustness and/or disturbance rejection. In Chapters 3 and 4 the MPC family of control algorithms and the proposed modification of the DMC, respectively, will be discussed. Then, in Chapters 5 and 6, the NLFSP will be introduced for systems with input nonlinearities and for the general stable nonlinear systems, respectively. And, finally, the conclusions are given in Chapter 7. The proofs of the theorems used throughout this thesis are presented in Appendix A, Appendix B has a minor discussion about the impulse response of filters, and a brief introduction to Input-to-State Stability (ISS) is done in Appendix C.

2 THE FILTERED SMITH PREDICTOR

Dead Time Compensators (DTC) are typical control strategies used to efficiently control dead-time processes. DTC schemes include a model of the process in the structure of the controller in order to cope with the dead time. The Smith Predictor (SP), arguably the first DTC, presented at the end of the 50's [5], was used to improve the performance of classical controllers (PI or PID controllers) for plants with dead time. The original SP is simple to understand and tune, and because of this, it is the best known and most widely used algorithm for dead-time compensation in industry. The main idea of the SP is to separate the controller in two parts, a closed-loop predictor and a primary controller in such a way that if the predictor model is a perfect representation of the process, the primary controller tuned for the dead-time free model can be used in the SP to control the dead-time process. However, it has some drawbacks and, over the past 30 years, numerous extensions and modifications of the SP have been proposed in order to: (a) improve the regulatory capabilities of the SP for measurable or unmeasurable disturbances; (b) to allow its use with unstable plants; (c) to improve the robustness or (d) to facilitate the tuning in a 2DOF structure [34]. See [3, 4] for a review of the SP and its modifications.

Most of the structures recently proposed in literature are more complex than the original SP and are specially proposed for one type of process model: stable, integrative or unstable. In some cases the solutions are only valid for simple models like first order plus dead time (FOPDT) or second order plus dead time (SOPDT) models. Robust 2DOF structures have been introduced in [35] only for stable and integrative processes. Other structures, including a disturbance-observer in the DTC, are presented in [36] for integrative processes and also analyzed in [37, 38]. A modified SP for the integrative case was presented in [39] and a simpler solution for the same case based on the estimation of the disturbance was presented in [40, 41]. Modified versions of this structure were presented in [42–45] where more complex algorithms and tuning rules are described. Other solutions for the unstable process case were presented in [46–51]. In particular, in [50] a 2DOF structure is proposed based on four controllers and in [48, 49] the structure has three controllers. A recent work analyzes the FOPDT unstable process case [52] and proposes a simpler tuning rule but the DTC is also based on a three-controller structure.

Because of implementation problems, only the discrete versions

of the dead time compensators are used in practice. However most of the papers do not pay attention to these issues. Some of the particular properties of the digital version of the SP and DTCs are discussed in [3, 53–55].

A simple and unified solution for controlling dead time processes, stable or unstable, is the Filtered Smith Predictor (FSP). The FSP was presented in [56] for stable cases, and extended to include integrating and unstable cases in [3, 34]. Its structure is based on a simple modification of the SP and both the designing and tuning of the controller are simple. Hence this thesis makes extensive use of the FSP structure, which maintains the simplicity of the original SP, and, with simple tuning guidelines, can provide the following advantages:

1) improved closed-loop robustness;

2) improved unmeasurable disturbance rejection capabilities.

However, it must be noted that there is a trade-off between the disturbance rejection capabilities and the closed-loop robustness, i.e., improving robustness generally makes the disturbance rejection response slower.

The rest of this chapter will provide an introduction to the FSP structure which is essential to understand the results obtained in this thesis.

2.1 THE FILTERED SMITH PREDICTOR STRUCTURE

The discrete FSP structure is shown in Figure 1. As can be seen the structure is the same as in the SP with two additional filters. $F(z)$ is a traditional reference filter to improve the set-point response and $F_r(z)$ is a predictor filter used to improve the predictor properties. The signals $r(t)$, $q(t)$, $n(t)$, $y_n(t)$, $e_p(t)$ and $y_p(t)$ are, respectively, the desired set-point, the input disturbance, the output disturbance, the output of the nominal model used by the predictor, the prediction error, and the delay-free prediction of the process. This structure has been proposed in [56] for FOPDT stable processes to improve the robustness of the traditional SP. Note that, if $F_r(z) = F(z) = 1$, the FSP is reduced to the classical SP [34].

Because of its characteristics, the FSP can be used to compute a controller taking into account the robustness, coping with unstable plants, improving the disturbance rejection properties, and decoupling the set-point and disturbance responses. Therefore, all the drawbacks

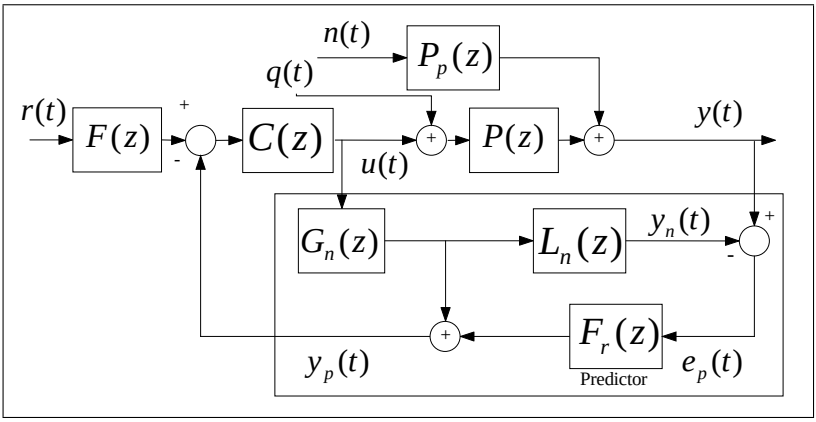

Figure 1 – Structure of the FSP.

of the SP are considered in the design, using only one structure, as was shown in [34].

In this structure, $P_n(z) = G_n(z)L_n(z)$, with $L_n(z) = z^{-d_n}$, is a model of the process, $G_n(z)$ is the dead-time-free model, which is also sometimes called fast model, $C(z)$ is the primary controller and $P_p(z)$ represents the dynamics of the disturbance in the output. The nominal closed-loop setpoint-to-output transfer function (when the model of the plant is perfect, $P(z) = P_n(z)$) is:

$$H_r(z) = \frac{Y(z)}{R(z)} = \frac{C(z)G_n(z)}{1 + C(z)G_n(z)} L_n(z)F(z). \tag{2.1}$$

Note that this is the classical closed-loop transfer function for delay-free systems with the exception that the delay appears in the numerator of the system. So, one advantage of the predictor is that, in the nominal case, the closed-loops poles of the system are independent of the process delay.

However, for the disturbance rejection, to better understand what happens it is best to analyze the FSP structure through the classical two-degree-of-freedom (2DOF) structure shown in Figure 2. To do this, first apply the Z-Transform to the prediction signal $y_p(t)$ in the FSP structure:

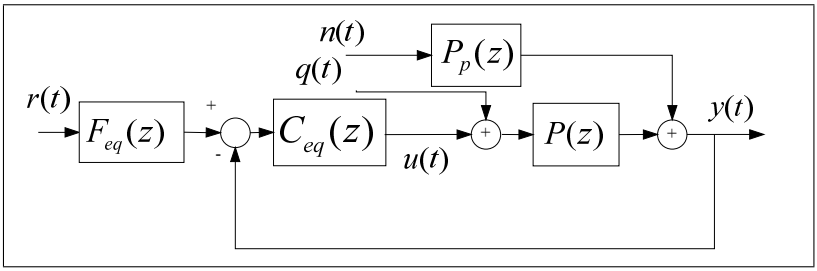

Figure 2 – Classical 2DOF control structure.

$$\begin{aligned} Y_p(z) =& G_n(z)U(z) + F_r(z)(Y(z) - G_n(z)L_n(z)U(z)) \\ =& S_n(z)U(z) + F_r(z)Y(z), \end{aligned} \tag{2.2}$$

with $S_n(z) = G_n(z)(1 - F_r(z)L_n(z))$.

Using this last equation the control signal $U(z)$ is given by

$$\begin{aligned} U(z) =& C(z)(F(z)R(z) - Y_p(z)) \\ =& C(z)(F(z)R(z) - G_n(z)(1 - F_r(z)L_n(z))U(z) - F_r(z)Y(z)) \\ U(z) =& \frac{C(z)F_r(z)}{1 + C(z)G_n(z)(1 - F_r(z)L_n(z))}(F_r^{-1}(z)F(z)R(z) - Y(z)). \end{aligned}$$

Comparing this last equation with the classical control loop, the following equivalent controller, $C_{eq}(z)$, and equivalent reference filter, $F_{eq}(z)$, are obtained:

$$C_{eq}(z) = \frac{C(z)F_r(z)}{1 + C(z)G_n(z)(1 - F_r(z)L_n(z))}, \tag{2.3}$$

$$F_{eq}(z) = F_r^{-1}(z)F(z). \tag{2.4}$$

Factoring the primary controller and the nominal plant in numerators and denominators, i.e., $C(z) = N_c(z)/D_c(z)$ and $G_n(z) = N_n(z)/D_n(z)$, the equivalent controller can be written as

$$C_{eq}(z) = \frac{D_n(z)N_c(z)F_r(z)}{D_c(z)D_n(z) + N_c(z)N_n(z)(1 - F_r(z)L_n(z))}.$$

Notice how the poles of the nominal model (the roots of $D_n(z)$)

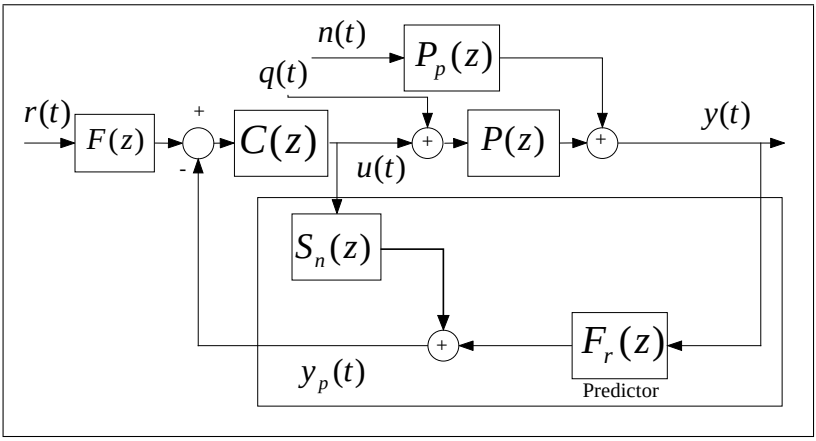

Figure 3 – Implementation Structure of the FSP.

are in the numerator of the equivalent controller. Consider first that $F_r(z)$ is not used, that is, $F_r(z) = 1$, then the classical SP is obtained. In this case, these zeros can not be eliminated from $C_{eq}(z)$ for any tuning of $C(z)$. This means that $C_{eq}(z)$ cancels all the open-loop poles of $G_n(z)$, thus, as it is well known in literature, these canceled poles will appear in the input-disturbance rejection response. The cancellation of the open-loop poles by the equivalent controller generates some basic problems: (i) if the system is open-loop stable, the disturbance rejection will always be at least as slow as the open-loop dynamics, and (ii) if the system is open-loop unstable, i.e., it has at least one pole outside the unit circle, the input disturbance rejection will also be unstable; (iii) in the particular case of one integrating pole (root at $z = 1$), the system is closed-loop stable but the input disturbance rejection response have an static error for constant disturbances, that is, $e(t) > 0$ for a very large t. These are three important draw-backs of the SP.

Also, another problem with open-loop unstable systems is that the predictor will be internally unstable because the prediction signal transfer function has the open-loop poles. However, note that in the FSP the prediction filter $F_r(z)$ appears in the denominator of $C_{eq}(z)$ (Eq. (2.3)). Hence, if tuned adequately, it can be used to cancel the unstable (or slow) zeros of the equivalent controller. Simply note that this can be obtained if the term $1 - F_r(z)L(z)$ has these undesirable roots. Therefore, as the equivalent controller will not cancel these poles anymore, they will not appear in the disturbance rejection response. On

the other hand this will turn the predictor internally stable as $S_n(z) = G_n(z)(1 - F_r(z)L(z))$ will have only stable poles. The implementation in this case must be done using the structure shown in Figure 3, where $S_n(z)$ is explicitly used.

The most important conceptual aspect of the FSP is that in the nominal case the predictor filter $F_r(z)$ only affects the disturbance rejection. This is expected since, without disturbances, the prediction is perfect, i.e., the prediction d_n samples ahead is equal to the output after the dead time, thus there is no prediction error.

As will be shown in Section 2.2, $F_r(z)$ can be used to improve the robustness or the disturbance rejection capabilities of the system without affecting the nominal set-point response. It can also be tuned to obtain an internally stable system when controlling unstable plants. Therefore, the proposed controller has enough degrees of freedom to obtain a compromise between robustness and the desired set-point and disturbance rejection responses.

2.2 TUNING PROCEDURE

The controller tuning is analyzed in two steps. First the nominal performance specifications are obtained, then the robustness is considered. Again here, as in the SP, the idea is to tune a primary controller based on the dead-time free model and to apply this controller to the system composed by the dead-time process and the predictor.

2.2.1 Nominal Performance

The correct tuning of $C(z)$, $F(z)$ and $F_r(z)$ must result in an internally stable nominal closed-loop system with the desired transfer functions between the set-point and the output and the disturbance and the output. Only the case of the input disturbance $q(t)$ will be considered, but the results can easily be extended to the output disturbance $n(t)$.

Given that $G_n(z) = N_n(z)/D_n(z)$ and $L_n(z) = z^{-d_n}$, the proposed primary controller $C(z) = N_c(z)/D_c(z)$ is tuned to allocate the closed-loop poles of the system, which are the roots of $D_{cl}(z)$. This is done solving the Diophantine equation:

$$D_{cl}(z) = N_n(z)N_c(z) + D_c(z)D_n(z)$$

where the polynomials $N_c(z), D_c(z)$ are obtained. Normally $D_c(z)$ includes a pole at $z = 1$ to obtain an offset-free closed-loop system[1]. This solution follows classical controller design [57], and gives a controller of the same order of the process model [34].

Thus, after defining the primary controller $C(z)$, the closed-loop transfer function is:

$$H_r(z) = F(z)\frac{N_n(z)N_c(z)z^{-d_n}}{D_{cl}(z)}.$$

To complete the tuning of the nominal closed-loop set-point to output response, the reference filter $F(z)$ is computed to correct the positions of any undesirable zeros present in $H_r(z)$.

Now, the nominal closed-loop transfer functions for input and output disturbances are:

$$H_q(z) = \frac{Y(z)}{Q(z)} = P_n(z)\left[1 - \frac{C(z)P_n(z)F_r(z)}{1 + C(z)G_n(z)}\right], \tag{2.5}$$

$$H_p(z) = \frac{Y(z)}{N(z)} = P_p(z)\left[1 - \frac{C(z)P_n(z)F_r(z)}{1 + C(z)G_n(z)}\right]. \tag{2.6}$$

From Eq. (2.5), note that since the primary controller $C(z)$ was already computed, the only degree of freedom left to modify the disturbance response is the prediction filter $F_r(z)$. Hence, $F_r(z)$ must be chosen to avoid the three main problems of the original structure proposed by Smith (the FSP for $F(z) = 1$ and $F_r(z) = 1$) [4, 34]:

- the disturbance rejection properties of the closed-loop system cannot be arbitrarily defined as the open-loop poles are also closed-loop poles of the transfer function $H_q(z)$;
- if the process has a pole p such that $|p| > 1$, the SP has an unstable closed-loop disturbance rejection response and the predictor is also internally unstable;
- for the particular case of integrative processes, the SP does not reject step disturbances at the plant input.

[1] The method to obtain the controller used here is the pole allocation, in which all the poles of $D_{cl}(z)$ are specified *a priori*. Other methods, for instance, the root-locus method, does not need the specification of all the closed-loop poles, requiring only that the closed-loop system is second-order dominant, for example. However, independent of the method chosen, $D_{cl}(z)$ can always be computed.

As stated in Section 2.1, these problems come from the fact that the equivalent controller of the SP structure, $C_{eq}(z)$ given by Eq. (2.3), cancels all the open-loop poles of the process. However, in the FSP the prediction filter $F_r(z)$ can be used to avoid this.

Comparing Eqs. (2.2) and (2.3), the equivalent controller can be describe as:

$$C_{eq}(z) = \frac{C(z)F_r(z)}{1 + C(z)S_n(z)}.$$

For simplicity, consider that $C(z)$ does not have common roots with $G_n(z)$. Hence, from this last equation, note that the open-loop poles of the process in the numerator of $C_{eq}(z)$ can only come from $S_n(z)$. Therefore, if $S_n(z)$ does not have these poles, they will not appear in the equivalent controller as zeros, avoiding the problems listed above [2].

From Eq. (2.2),

$$S_n(z) = G_n(z)(1 - z^{-d_n}F_r(z)).$$

So, in order to remove the undesirable poles from $S_n(z)$ that comes from $G_n(z)$, the following condition must be satisfied

$$(1 - z^{-d_n}F_r(z))\big|_{z=z_i} = 0, \tag{2.7}$$

where z_i is the location of the ith undesirable root. This means that this term of $S_n(z)$ must have zeros at all the undesired (slow/unstable) pole locations of $G_n(z)$.

Using $G_n(z) = N_n(z)/(D_n^-(z)D_n^+(z))$, where $D_n^-(z)$ represents the undesired open-loop poles location, and $F_r(z) = N_f(z)/D_f(z)$, then

$$\begin{aligned} S_n(z) =& \frac{N_n(z)}{D_n^-(z)D_n^+(z)} \left(1 - z^{-d_n}\frac{N_f(z)}{D_f(z)}\right) \\ =& \frac{N_n(z)}{D_n^-(z)D_n^+(z)} \left(\frac{D_f(z) - z^{-d_n}N_f(z)}{D_f(z)}\right), \end{aligned}$$

and given the condition represented by Eq. (2.7),

$$D_f(z) - z^{-d_n}N_f(z) = D_n^-(z)N_s(z),$$

where $N_s(z)$ represents the remaining zeros of the left-hand-side poly-

[2] Even if $C(z)$ has common roots with $G_n(z)$ it can be proven that these results are also true [3].

nomial. Hence, $S_n(z)$ is reduced to

$$S_n(z) = \frac{N_n(z)}{D_n^+(z)} \frac{N_s(z)}{D_f(z)}.$$

Using this procedure, $S_n(z)$ will not have the undesired open-loop poles which will, in turn, avoid the cancellation of these poles by the equivalent controller. The closed-loop transfer function of the input disturbance[3] will be analyzed to verify this statement. From Eq. (2.5)

$$\begin{aligned} H_q(z) =& G_n(z)z^{-d_n}\left[1 - \frac{C(z)G_n(z)z^{-d_n}F_r(z)}{1 + C(z)G_n(z)}\right] \\ =& G_n(z)z^{-d_n}\left[\frac{1 + C(z)G_n(z) - C(z)G_n(z)z^{-d_n}F_r(z)}{1 + C(z)G_n(z)}\right] \\ =& G_n(z)z^{-d_n}\left[\frac{1 + C(z)S_n(z)}{1 + C(z)G_n(z)}\right]. \end{aligned}$$

Using the factorization of $C(z)$, $G_n(z)$, $S_n(z)$,

$$H_q(z) = \frac{N_n(z)z^{-d_n}}{D_n^-(z)D_n^+(z)}\left[\frac{D_c(z)D_n^-(z)D_n^+(z) + N_c(z)D_n^-(z)N_n(z)N_s(z)}{D_{cl}(z)D_f(z)}\right],$$

and since $D_n^-(z)$ is a factor in both the numerator and denominator of this transfer function, it can be removed,

$$H_q(z) = \frac{N_n(z)z^{-d_n}}{D_n^+(z)}\left[\frac{D_c(z)D_n^+(z) + N_c(z)N_n(z)N_s(z)}{D_{cl}(z)D_f(z)}\right],$$

which is exactly the desired result of the tuning procedure.

Some notes:

- the prediction filter $F_r(z)$ must have unitary gain, i.e., $F_r(1) = 1$, otherwise, $H_q(1) \neq 0$, and there will be an static error at steady state;

- it is considered that $D_{cl}(z)$ already has the desired poles for the disturbance rejection. If this is not the case, $F_r(z)$ must also be tuned to cancel some or all the poles of $D_{cl}(z)$;

- the poles of $F_r(z)$ will also be closed-loop poles of the disturbance rejection, thus, these poles must be faster than the ones being

[3] A similar analysis can be done to the output disturbance.

cancelled, otherwise, there will not be any improvement in the disturbance rejection response.

- a more complete set of tuning guidelines for the prediction filter $F_r(z)$ can be found in [34].

2.2.2 Robust Stability

To analyze the robustness consider a family of plants $P(z)$ such that $P(z) = P_n(z)[1 + \delta P(z)]$. The closed-loop characteristic equation for $P(z)$ is then:

$$1 + C(z)G_n(z) + C(z)P_n(z)F_r(z)\delta P(z) = 0.$$

Considering that the nominal system is stable, the robust stability for the FSP can be derived from Nyquist stability criterion (a special case of the small-gain theorem for linear systems) [58]:

$$|\delta P(z)| < |dP(z)| = \left| \frac{1 + C(z)G_n(z)}{C(z)G_n(z)} \right| \frac{1}{|F_r(z)|}, \tag{2.8}$$

where $z = e^{-j\omega T_s}$, $\forall \omega \in [0, \pi/T_s)$, with T_s being the sampling time.

For a given nominal reference to output response, the prediction filter $F_r(z)$ can be used to shape the value of $dP(z)$, hence affecting the robustness condition.

Consider that $F_r(z) = \frac{N_o(z)}{D_o(z)}$ and that $D_o(z) = (z - z_o)^{n_o}$, then

$$|dP(z)| = \left| \frac{1 + C(z)G_n(z)}{C(z)G_n(z)} \right| \frac{|D_o(z)|}{|N_o(z)|}. \tag{2.9}$$

Note that higher values of z_o give higher values of $|dP(z)|$, increasing the robustness and at the same time giving a slower disturbance rejection response. If the plant is open-loop stable, in order to achieve the robust stability condition z_o can be increased as long as $|z_o| < 1$ (to maintain the stability of the filter). On the other hand, increasing z_o may result in a slow closed-loop system because the filter dynamics has an impact on disturbance rejection [34].

If the plant is open-loop unstable case, robustness cannot be increased arbitrarily. This is an expected result because certain feedback action is needed to maintain stability and thus, the detuning of the controller (with high values of z_o) has a limit [1]. That is, there is an

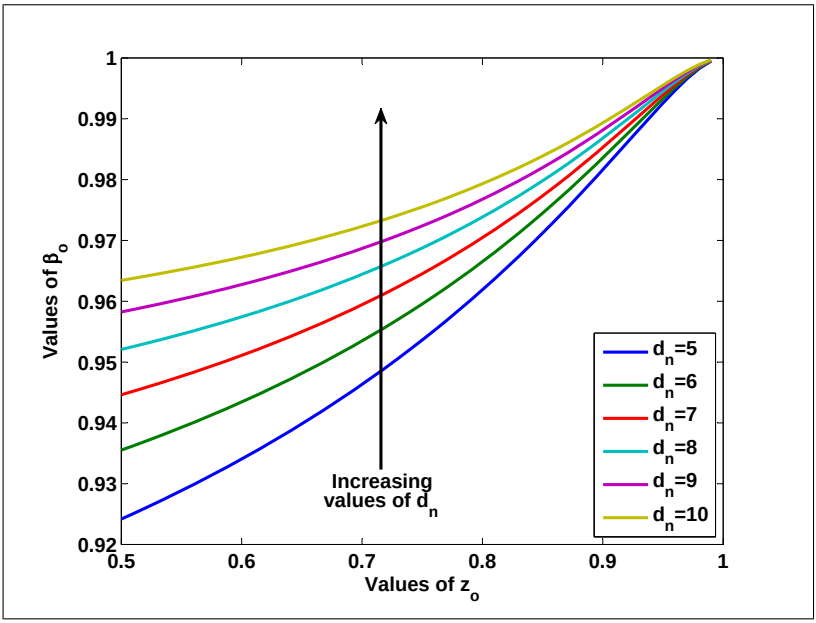

Figure 4 – Relation between β_o, z_o and d_n for $z_u = 1.1$.

achievable robustness for unstable plants as will be illustrated next. This result is valid for all unstable processes, however, for the sake of clarity, a model with only one pole outside the unit circle will be used in the following analysis.

Assume that the process model is $G_n(z) = \frac{K}{z - z_u}$, where $z_u > 1$. Consider that $C(z)$ is chosen for a stable $H_r(z)$ with unitary static gain and the desired closed loop $D_{cl}(z)$. Defining $F_r(z)$ as

$$F_r(z) = \frac{N_o(z)}{D_o(z)} = \frac{K'(z - \beta_o)}{(z - z_o)^2},$$

where z_o is the tuning parameter, $K' = (1 - z_o)^2/(1 - \beta_o)$, and β_o has to verify

$$\beta_o = \frac{(z_u - z_o)^2 z_u^{d_n} - z_u(1 - z_o)^2}{(z_u - z_o)^2 z_u^{d_n} - (1 - z_o)^2}, \tag{2.10}$$

to eliminate the effect of the open-loop unstable pole at $z = z_u$.

It is not easy to see how β_o varies with different values of d_n and z_o through the analysis of Eq. (2.10). Thus, for $z_u = 1.1$, the values

of β_o for different values of z_o and d_n were computed, and they can be seen in the plot in Figure 4. For a fixed d_n, as z_o increases, so does β_o, which is always bigger than z_o. And, fixing z_o, as d_n increases, so does β_o, and at the limit, β_o tends to 1, as was shown in [59] and in Figure 4.

The robustness of the controller can be analyzed using the admissible plant uncertainty bound given by Eq. (2.9). This expression shows that higher values of z_o give, in general, higher values of $dP(z)$, however, β_o also increases with z_o, as was shown in Figure 4, and may reach high values, principally when d_n is big; thus, $dP(z)$ cannot be increased arbitrarily. This shows the limitation imposed by the dead time and the unstable time constant of the system. For an unstable dead-time dominant system, the closed-loop system can become unstable with an infinitesimal value of modelling error [3].

2.3 EXAMPLES

To illustrate the tuning procedures of the FSP, two simple first order plus dead time (FOPDT) examples will be studied, a stable and an unstable process.

2.3.1 Stable FOPDT models

In the stable case the tuning of the FSP is simple and intuitive as the correct tuning of $C(z)$ gives an internally stable closed-loop system. Thus, $F_r(z)$ is used only to improve the robustness or disturbance rejection performance of the system. These two specifications cannot be achieved simultaneously in the same range of frequencies; that is, there is a trade-off between robustness and performance. Furthermore, it is important to note that the elimination of the open-loop poles from $H_q(z)$ only has an important effect if the desired closed-loop poles (the roots of $D_{cl}(z)$) are faster than the open-loop ones. This is not the case when the process has a dominant dead time, because slow responses are necessary to achieve robust performance of the closed-loop system [4, 56]. Furthermore, when the dead time is dominant the contribution of the open-loop poles to the closed-loop response will be small, thus their elimination will contribute with a small increment in the speed of the transients.

Consider a First Order Plus Dead Time Model (FOPDT) des-

cribed by

$$P_n(z) = \frac{0.1}{z - 0.95} z^{-5},$$

an the following PI controller

$$C(z) = \frac{(z - 0.95)}{z - 1},$$

the sampling time is $T_s = 1$ [s]. Since the zero of the controller is the same as the pole of the plant, the reference closed-loop transfer function is also an FOPDT

$$\frac{C_n(z)P_n(z)}{1 + C_n(z)G_n(z)} = \frac{0.1z^{-5}}{z - 0.9},$$

with settling time of approximately 35 [s].

Initially, the prediction filter $F_r(z)$ is tuned to consider only the improvement of the disturbance rejection response. Hence, this filter is defined as

$$F_r(z) = \frac{(z - 0.9)}{0.1} \frac{K'(z - \beta_o)}{(z - z_o)^2},$$

which will completely decouple the reference response from the disturbance one because the filter cancels the poles of the reference closed-loop transfer function.

The value of z_o is chosen as 0.75 to obtain a fast disturbance rejection. Hence, $\beta_o = 0.9009$ to cancel the open-loop pole at $z = 0.95$, (obtained from Eq. (2.10)). Then filter $F_r(z)$ is

$$\begin{aligned} F_r(z) &= \frac{(z - 0.9)}{0.1} \frac{(1 - 0.75)^2(z - 0.9009)}{(1 - 0.9009)(z - 0.75)^2} \\ &= \frac{6.3098(z - 0.9009)(z - 0.9)}{(z - 0.75)^2}. \end{aligned}$$

The nominal response of the system with and without this filter can be seen in Figure 5. An input step disturbance with 0.5 amplitude is applied at time 60 [s]. It is very noticeable the improvement in the disturbance rejection with the prediction filter since the open-loop pole was removed from the nominal response.

Now, consider that the time delay is uncertain, i.e., the real value of the the dead time is $d = d_n \pm 2$ samples. Then the robustness condition given by Eq. (2.8) must be analyzed to guarantee that the closed-loop system is stable. In Figure 6, it can be seen that this

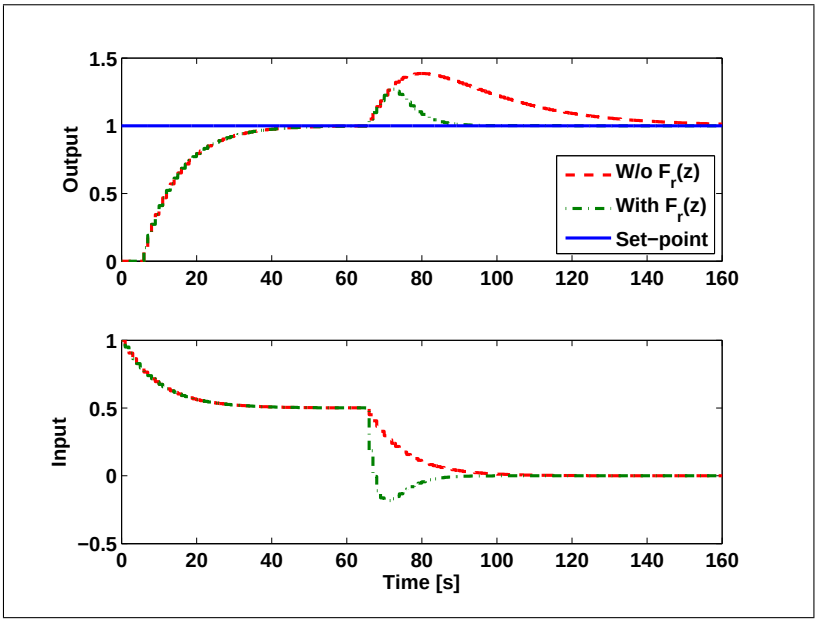

Figure 5 – Nominal response of the stable SISO example with, and without filter $F_r(z)$.

condition is not satisfied, hence robustness is not guaranteed, which is clear through the simulation results in Figure 7. Therefore, the filter parameter z_o must be tuned again so as to satisfy the robustness condition. This is achieved with $z_o = 0.85$, which gives $\beta_o = 0.9238$ and the filter

$$F_{r2}(z) = \frac{2.9524(z - 0.9238)(z - 0.9)}{(z - 0.85)^2}.$$

With this prediction filter, the robustness condition is satisfied, as it is shown in Figure 6. The simulation results are shown in Figure 7. With $F_{r2}(z)$ the closed-loop system becomes stable even in the presence of uncertainties, although with a slower disturbance rejection, which is the price to pay.

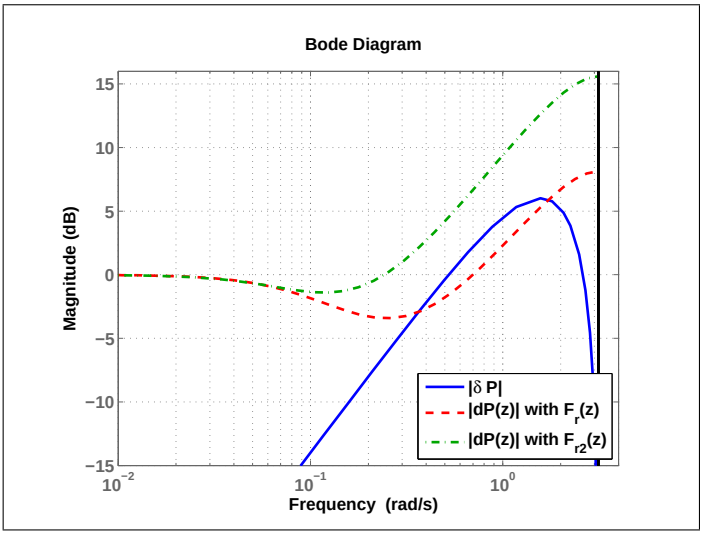

Figure 6 – Robustness condition for the filters $F_r(z)$ and $F_{r2}(z)$.

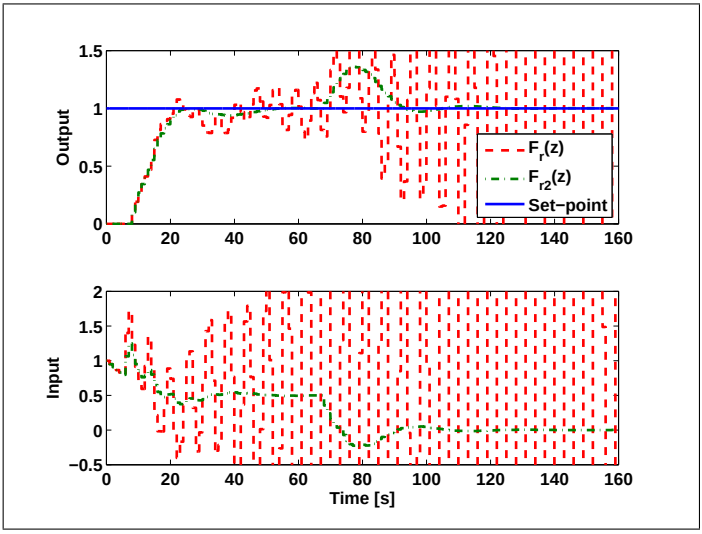

Figure 7 – Simulation results for the stable SISO case with uncertainties.

2.3.2 Unstable FOPDT models

A PI controller $C(z) = \frac{K_c(z-b)}{z-1}$ is computed to define the set-point response. Using $G_n(z) = \frac{K}{z-a}$, with $|a| > 1$, the nominal closed-loop transfer function can be chosen as:

$$H_r(z) = \frac{K_c K(z-b)}{(z-z_r)^2} z^{-d_n},$$

where z_r is the desired closed-loop pole and $K_c K(1-b)/(1-z_r)^2 = 1$. Also, a reference filter

$$F(z) = \frac{(1-b)(z-b_f)}{(1-b_f)(z-b)},$$

can be used to eliminate the effect of the controller zero.

Considering $T_s = 1$ [s], $a = 1.5$, $K = 1$ and $d_n = 3$ and $z_r = 0.95$, which gives a settling time of 90 [s], the parameters of the controller $C(z)$ are $K_c = 0.6$ and $b = 0.9958$. The zero of the set-point filter is chosen as $b_f = 0.8$.

For the disturbance rejection, following the same steps as in the previous example, where the prediction filter $F_r(z)$ completely decouples the reference and the disturbance responses, the remaining filter parameters are chosen to cancel the unstable root at $z = a$. Using Eq. (2.10) with $z_o = 0.9$ results in $\beta_o = 0.9959$. Then filter $F_r(z)$ is

$$\begin{aligned} F_r(z) &= \frac{(z-z_r)^2}{K_c K(z-b)} \frac{(1-z_o)^2(z-\beta_o)}{(1-\beta_o)(z-z_o)^2} \\ &= \frac{4.0167(z-0.95)^2(z-0.9959)}{(z-0.9958)(z-0.9)}. \end{aligned}$$

The simulation results are presented in Figure 8, where it can be seen that the FSP stabilizes the nominal process and rejects input disturbances. Also note that in the unstable case, the implementation structure shown in Figure 3 must be used.

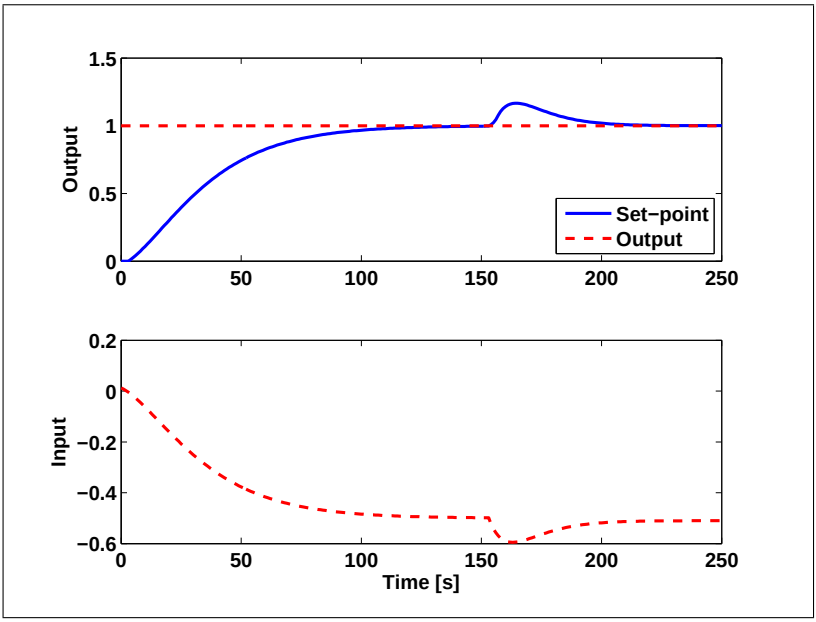

Figure 8 – Simulation results for the nominal unstable SISO.

2.4 MULTIVARIABLE CASE

In this section, based on [1, 60, 61], the FSP concept will be extended to multivariable processes. Using the same structure presented in Figure 1, consider a MIMO system with m inputs, n outputs and p output disturbances, $\mathbf{n}(t)$ and $\mathbf{q}(t)$ are vectors that represent, respectively, output and input disturbances, $\mathbf{F_r}(z)$ is a $n \times n$ matrix transfer function which will be tuned to modify the robustness and disturbance rejection properties of the system, $\mathbf{r}(t)$ represent the set-points, $\mathbf{C}(z)$ is a $m \times n$ controller tuned to stabilize the system's nominal fast model $\mathbf{G_n}(z)$ (the full nominal system model is given by $\mathbf{P_n} = \mathbf{L_n}\mathbf{G_n}$, in which $\mathbf{L_n}$ is related to the time delay). $\mathbf{P_p}(z)$ represents how $\mathbf{n}(t)$ affects $\mathbf{y}(t)$ and the plant is defined as

$$\mathbf{P}(z) = \begin{bmatrix} G_{11}(z)z^{-d_{11}} & \cdots & G_{1m}(z)z^{-d_{1m}} \\ G_{21}(z)z^{-d_{21}} & \cdots & G_{2m}(z)z^{-d_{2m}} \\ \vdots & \ddots & \vdots \\ G_{n1}(z)z^{-d_{n1}} & \cdots & G_{nm}(z)z^{-d_{nm}} \end{bmatrix}.$$

In this model, $G_{ij}(z)z^{-d_{ij}}$ is the transfer function relating the jth input with the ith output where $G_{ij}(z)$ is a delay-free transfer function and d_{ij} is the discrete dead time. The effective dead time of each output i is d_i, computed as the minimal delay of the ith row, i.e., $d_i = \min_{j=1\ldots n}(d_{ij})$. Thus, defining $\mathbf{L}(z) = diag\{z^{-d_1}, \ldots, z^{-d_n}\}$ as the MIMO delay of the plant $\mathbf{P}(z)$ and $\mathbf{G}(z)$ as the model without the common dead-times (also called fast model), it follows that $\mathbf{P}(z) = \mathbf{L}(z)\mathbf{G}(z)$. It is worth noting that $\mathbf{G}(z)$ may still contain multiple delays [1]. The controller $\mathbf{C}(z)$ is given by

$$\mathbf{C}(z) = \begin{bmatrix} C_{11}(z) & C_{21}(z) & \cdots & C_{n1}(z) \\ C_{12}(z) & C_{22}(z) & \cdots & C_{n2}(z) \\ \vdots & \vdots & \ddots & \vdots \\ C_{1m}(z) & C_{2m}(z) & \ldots & C_{nm}(z) \end{bmatrix},$$

where $C_{ij}(z)$ is a transfer function representing the controller from the jth input to the ith output.

Given the SISO FSP structure presented in Figure 1, the only difference for the MIMO case is the substitution of the SISO transfer functions for MIMO matrix transfer functions. Similarly to the SISO case, it can be proven that the closed-loop relations in the nominal case ($\mathbf{P} = \mathbf{P_n}$) are given by[4]

$$\mathbf{H_R}(z) = \mathbf{P_n}\mathbf{C}(\mathbf{I} + \mathbf{G_n}\mathbf{C})^{-1} \quad (2.11)$$

$$\mathbf{H_N}(z) = \{\mathbf{I} - \mathbf{P_n}\mathbf{C}(\mathbf{I} + \mathbf{G_n}\mathbf{C})^{-1}\mathbf{F_r}\}\mathbf{P_p} \quad (2.12)$$

$$\mathbf{H_Q}(z) = \{\mathbf{I} - \mathbf{P_n}\mathbf{C}(\mathbf{I} + \mathbf{G_n}\mathbf{C})^{-1}\mathbf{F_r}\}\mathbf{P_n} \quad (2.13)$$

where $\mathbf{I}$ is an identity matrix of appropriate dimensions.

Note that the same properties for the SISO case are valid here, the closed-loop nominal reference response does not depend on the prediction filter $\mathbf{F_r}(z)$, but the filter affects the disturbance rejection, hence, it can be tuned to improved it. The filter also affects the MIMO robustness as will be seen in section 2.4.2.

2.4.1 MIMO Filter design

In this section only the internal stability of the predictor will be addressed. A detailed explanation of tuning procedures to improve

[4]The dependance on the complex variable z was omitted for visualization purposes

disturbance rejection in the MIMO case can be found in [1].

Consider that the primary controller $\mathbf{C}(z)$ was designed to stabilize $\mathbf{G_n}(z)$ and also to fulfill some requirements concerning the closed-loop system behavior (note that $\mathbf{C}(z)$ must guarantee the internal stability of the fast model $\mathbf{G_n}(z)$ or the delayed closed-loop system will not be internally stable). This can be reached by any classical MIMO control design approach. In general, integral action is included in $\mathbf{C}(z)$ to obtain an offset-free closed-loop response [1].

The proposed predictor filter $\mathbf{F_r}(z)$ is an $n \times n$ diagonal filter with diagonal elements $F_{ri}(z)$ which are tuned similarly to the SISO case, i.e., they will be chosen considering the robustness and disturbance characteristics of the closed-loop system, and the internal stability of the predictor. Also, as in the SISO case, considering that $\mathbf{C}(z)$ has integral action, the prediction filter must satisfy $\mathbf{F_r}(1) = \mathbf{I}$, otherwise $\mathbf{H_Q}(1) \neq 0$, i.e., the error will not be zero, as can be verified from Eq. (2.13).

It is interesting to note that the feedback signal $\mathbf{y_p}(t)$ produced by the predictor is given by [1]

$$\begin{aligned}\mathbf{Y_p}(z) &= \mathbf{G_n}(z)\mathbf{U}(z) + \mathbf{F_r}(z)[\mathbf{Y}(z) - \mathbf{P_n}(z)\mathbf{U}(z)] \\ &= \mathbf{F_r}(z)\mathbf{Y}(z) + [\mathbf{G_n}(z) - \mathbf{F_r}(z)\mathbf{P_n}(z)]\mathbf{U}(z) \\ &= \mathbf{F_r}(z)\mathbf{Y}(z) + \mathbf{S_n}(z)\mathbf{U}(z), \end{aligned} \tag{2.14}$$

with $\mathbf{S_n}(z) = (\mathbf{I} - \mathbf{F_r}(z)\mathbf{L_n}(z))\mathbf{G_n}(z)$.

As in the SISO case, if the process is stable and $\mathbf{C}(z)$ stabilizes $\mathbf{G_n}(z)$, then the FSP is internally stable, and the predictor filter design will consider only robustness and disturbance rejection specifications. If $\mathbf{G_n}(z)$ is open-loop unstable, other conditions must be satisfied to guarantee the internal stability of the FSP.

If $\mathbf{C}(z)$ stabilizes $\mathbf{G_n}(z)$, for the internal stability it is sufficient that the predictor is stable, i.e., Eq. (2.14) does not have poles outside the unit circle [1].

Since, by definition $\mathbf{F_r}(z)$ is stable, only the stability of $\mathbf{S_n}(z)$ remains. Given that $\mathbf{F_r}(z)\mathbf{L_n}(z) = \mathbf{L_n}(z)\mathbf{F_r}(z)$, because both matrices are diagonal, each row of $\mathbf{S_n}(z)$ can be computed as

$$\begin{aligned}\mathbf{S_{ni}}(z) = \big[&[1 - F_{r_i}(z)z^{-d_i}]G_{i1}(z) \quad [1 - F_{r_i}(z)z^{-d_i}]G_{i2}(z) \quad \cdots \\ &[1 - F_{r_i}(z)z^{-d_i}]G_{in}(z)\big] ,\end{aligned}$$

where z^{-d_i} is the ith diagonal element of $\mathbf{L_n}(z)$, that corresponds to the minimal delay of the ith row of the model of the plant.

To cancel the undesired roots of the denominator in all the transfer functions of this row, $[1 - F_{r_i}(z)z^{-d_i}]$ must have zeros in all these roots. This is satisfied if $F_{r_i}(z)$ meets the following condition:

$$\frac{d}{dz}\left(1 - F_{r_i}(z)z^{-d_i}\right)\Big|_{z=z_j} = 0, \tag{2.15}$$

where z_j is the jth undesirable roots of the ith line of $\mathbf{S_n}(z)$. This condition is equivalent to the one obtained in the SISO case (see Section 2.2.1), thus by polynomial division it is possible to implement the predictor equations associated to $\mathbf{S_{ni}(z)}$ without the undesirable poles and avoiding internal instability problems.

Notice that the proposed structure is able to deal with instabilities at any elements of the matrix transfer function (not just in the main diagonal) given that the primary controller is able to stabilize the fast model of the plant. As the filter is computed in order to eliminate all the undesirable poles of the transfer functions of a row, the prediction is stable even if the instabilities appear outside of the main diagonal [1].

2.4.2 Closed-loop Robustness

For robustness analysis, additive uncertainties are considered, i.e.

$$\mathbf{P}(z) = \mathbf{P_n}(z) + \mathbf{\Delta P}(z),$$

where $\mathbf{P}(z)$ represents the real plant, $\mathbf{P_n}(z)$ the nominal model, and $\mathbf{\Delta P}(z)$ the uncertainties. In general $\mathbf{\Delta P}(z)$ can be written as in Eq. (2.16)

$$\mathbf{\Delta P}(z) = \mathbf{W_2}(z)\mathbf{\Delta}(z)\mathbf{W_1}(z), \tag{2.16}$$

where $\overline{\sigma}(\mathbf{\Delta}(z)) \leq 1$ (or equivalently $\|\mathbf{\Delta}(z)\|_\infty < 1$) and, for this case, $\mathbf{\Delta}(z)$ is a full matrix, $\mathbf{W_1}(z)$ and $\mathbf{W_2}(z)$ are two stable matrix transfer functions that characterize the spatial and frequency structure of the uncertainty, and $\overline{\sigma}(\mathbf{X})$ denotes the maximum singular value of $\mathbf{X}$ [62][5].

[5]The maximum singular value of a $n \times m$ matrix $\mathbf{X}$, whose elements are a_{ij}, is given by

$$\|\mathbf{X}\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^{m} |a_{ij}|,$$

or the maximum between the absolute sum of the lines.

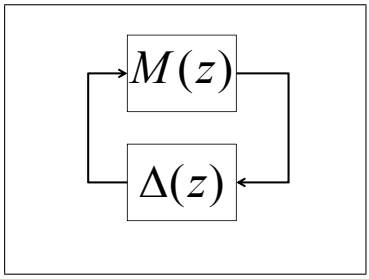

Figure 9 – $\mathbf{M}-\mathbf{\Delta}$ structure used for robustness analysis.

The $\mathbf{M}-\mathbf{\Delta}$ structure shown in Figure 9 is used to analyze the robustness. This structure can be obtained from the original system with $\mathbf{R}=\mathbf{0}$, $\mathbf{N}=\mathbf{0}$, $\mathbf{Q}=\mathbf{0}$ and

$$\mathbf{M}(z) = -\mathbf{W_1}(z)\mathbf{C}(z)[\mathbf{I}+\mathbf{G_n}\mathbf{C}(z)]^{-1}\mathbf{F_r}(z)\mathbf{W_2}(z),$$

or equivalently

$$\mathbf{M}(z) = -\mathbf{W_1}(z)\mathbf{M_0}(z)\mathbf{F_r}(z)\mathbf{W_2}(z),$$

where $\mathbf{M_0}(z) = \mathbf{C}(z)[\mathbf{I}+\mathbf{G_n}(z)\mathbf{C}(z)]^{-1}$ is stable because $\mathbf{C}(z)$ stabilizes $\mathbf{G_n}(z)$.

The robust stability condition, using the small-gain theorem, under block diagonal disturbances is given by [62]

$$\bar{\sigma}(\mathbf{\Delta}(z)) < \frac{1}{\overline{\sigma}(\mathbf{M}(z))},\ z = e^{j\omega T_s},\ \forall\omega \in [0, \pi/T_s).$$

Now, using $\overline{\sigma}(\mathbf{X}.\mathbf{Y}) \leq \overline{\sigma}(\mathbf{X})\overline{\sigma}(\mathbf{Y})$ it is possible to write

$$\overline{\sigma}(\mathbf{M}) \leq \overline{\sigma}(\mathbf{M_0})\overline{\sigma}(\mathbf{F_r})\overline{\sigma}(\mathbf{W_1})\overline{\sigma}(\mathbf{W_2}). \tag{2.17}$$

So, for given uncertainty matrices $\mathbf{W_1}$, $\mathbf{W_2}$, and a defined primary controller $\mathbf{C}(z)$, the robustness of MIMO-FSP is defined by $\overline{\sigma}(\mathbf{F_r})$.

As $\mathbf{F_r}(z)$ is diagonal, the singular values are the magnitudes of the elements of the diagonal, i.e.

$$\sigma_i(\mathbf{F_r}(z)) = \mid F_{r_i}(z) \mid,\ z = e^{j\omega T_s},\ \ \forall\omega \in [0, \pi/T_s).$$

Thus, the robustness characteristics of the controller are defined

by the shape of the scalar functions $F_{r_i}(z)$, similarly to the SISO case [1].

Note that, as in the SISO case, when controlling unstable processes, robustness can not be increased arbitrarily by tuning $F_{r_i}(z)$ because a minimal feedback action is necessary to maintain nominal closed-loop stability [3].

2.4.3 Example

The case to be considered will be the water-methanol distillation column, which was modeled in [63], and controlled with the MIMO-FSP structure in [1]. This is a typical MIMO plant with strong interaction and dead times. A simplified schematic representation of the process showing just the variables of interest is presented in Figure 10 and its transfer function matrices are given by

$$\mathbf{P}(s) = \begin{bmatrix} \dfrac{12.8\ e^{-s}}{16.7s+1} & \dfrac{-18.9\ e^{-3s}}{21s+1} \\ \dfrac{6.6\ e^{-7s}}{10.9s+1} & \dfrac{-19.4\ e^{-3s}}{14.4s+1} \end{bmatrix}, \quad \mathbf{P_p}(s) = \begin{bmatrix} \dfrac{3.8\ e^{-8.1s}}{14.9s+1} \\ \dfrac{4.9\ e^{-3.4s}}{13.2s+1} \end{bmatrix},$$

where $\mathbf{y}(s) = \mathbf{P}(s)\mathbf{u}(s) + \mathbf{P_p}(s)n(s)$, $\mathbf{y}(s) = [y_1(s)\ \ y_2(s)]^T$, $\mathbf{u}(s) = [u_1(s)\ \ u_2(s)]^T$, $y_1(s)$ is the overhead product mole fraction of methanol, $y_2(s)$ is the bottom product mole fraction of methanol, $u_1(s)$ is the reflux flow rate, $u_2(s)$ is the reboiler steam flow rate, and $n(s)$ is the feed flow rate. The dead times and the time constants have units of minutes, the mole fractions are given in percent, and the flow rates have units of pounds per minute ($1\,[\text{lb/min}] \approx 7.56 \cdot 10^{-3}\,[\text{kg/s}]$). The considered operating point is given by $y_1 = 96.25\,[\text{mol\%}]$, $y_2 = 0.50\,[\text{mol\%}]$, $u_1 = 1.95\,[\text{lb/min}]$, $u_2 = 1.71\,[\text{lb/min}]$, and $n = 2.45\,[\text{lb/min}]$.

The discrete representation of the model with a zero-order hold and a sampling time $T_s = 1\,[\text{min}]$ is given by Eq. (2.18). The disturbance model will not be discretized, since a disturbance feedforward action will not be considered. However, it is possible to add the feedforward action to the control structure.

$$\mathbf{P}(z) = \begin{bmatrix} \dfrac{0.7440z^{-1}}{z-0.9419} & \dfrac{-0.8789z^{-3}}{z-0.9535} \\ \dfrac{0.5786z^{-7}}{z-0.9123} & \dfrac{-1.3015z^{-3}}{z-0.9329} \end{bmatrix}. \tag{2.18}$$

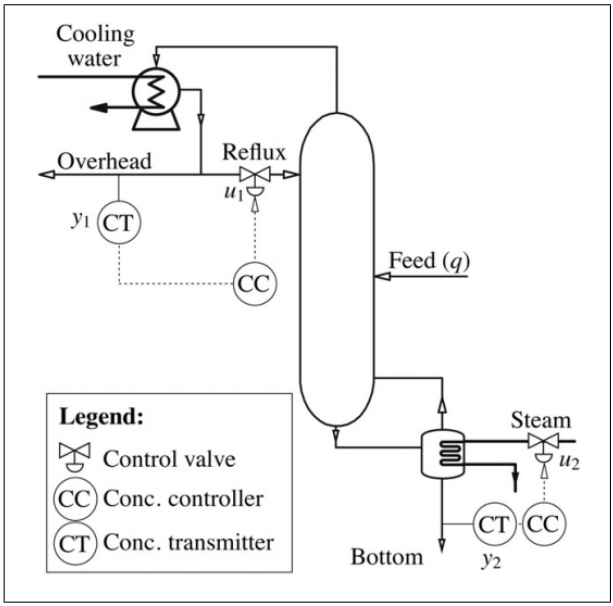

Figure 10 – Schematic representation of a water-methanol distillation column [1].

The minimal delays of each row are 1 [min] and 3 [min], respectively. Thus, the proposed structure will be able to compensate these delays and the main compensators may be tuned considering the fast model given by

$$\mathbf{G}(z) = \begin{bmatrix} \dfrac{0.7440}{z-0.9419} & \dfrac{-0.8789z^{-2}}{z-0.9535} \\ \dfrac{0.5786z^{-4}}{z-0.9123} & \dfrac{-1.3015}{z-0.9329} \end{bmatrix}.$$

In principle any control structure can be used and it may be tuned considering the fast model. In this case, the controller will be tuned following the case study presented in [1], i.e, a diagonal PI controller $\left(C_{jj}(z) = Kc_j + \frac{Kc_j}{Ti_j}\frac{T_s z}{z-1}, j = \{1,2\};\ C_{ij} = 0, i \neq j\right)$ with $K_{c1} = 0.5$, $T_{i1} = 9$, $K_{c2} = -0.2$, and $T_{i2} = 9$. The results are compared to the ones obtained using the traditional multivariable Smith predictor (MIMO-SP) with the same diagonal PI controller. No reference filter

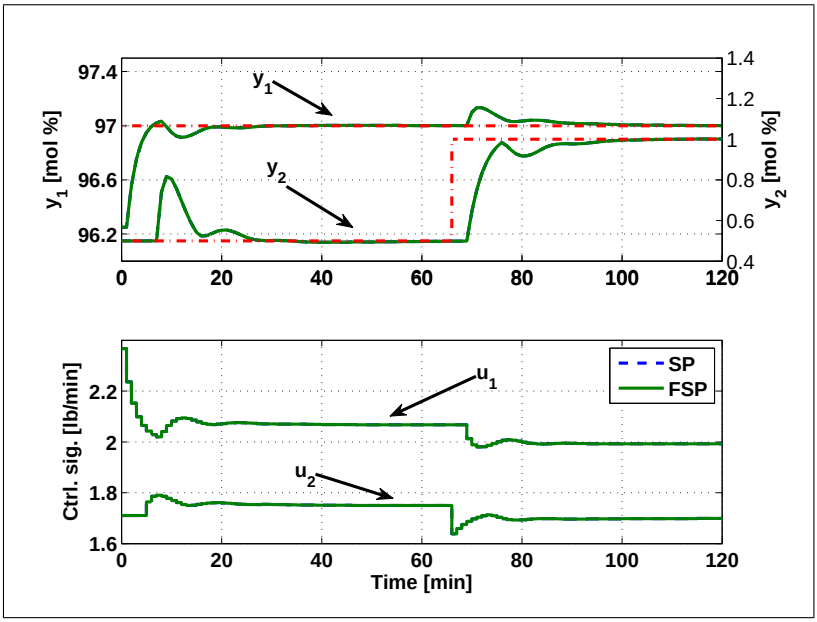

Figure 11 – Nominal set-point tracking for the MIMO case study.

is used in both cases, i.e., $\mathbf{F}(z) = \mathbf{I}$.

The filter $\mathbf{F_{r1}}(z)$ is tuned for the disturbance rejection to be performed in about 15 [min]. Thus, all the open-loop poles of the model of the plant that are outside the circle of radius $e^{-(3/15)T_s} \approx 0.8$ are cancelled by the predictor filter (Eq. (2.15)). The poles of the filter were chosen to be slightly faster than the desired dynamics ($z_{o1} = z_{o2} = 0.7$). Thus, the predictor filter is given by

$$\mathbf{F_{r1}}(z) = \begin{bmatrix} \frac{1.495(z^2-1.54z+0.6007)}{(z-0.7)^2} & 0 \\ 0 & \frac{2.376(z^2-1.645z+0.6827)}{(z-0.7)^2} \end{bmatrix}.$$

For the simulation, the reference of the overhead mole fraction (y_1) is increased by 0.75 [mol]% at $t = 0$ [min], the reference of the bottom product mole fraction (y_2) is increased by 0.5 [mol]% at $t = 66$ [min], and the feed flow rate (n) is decreased by 0.15 [lb/min] at $t = 132$ [min]. Simulation results are presented in Figure 11 (nomi-

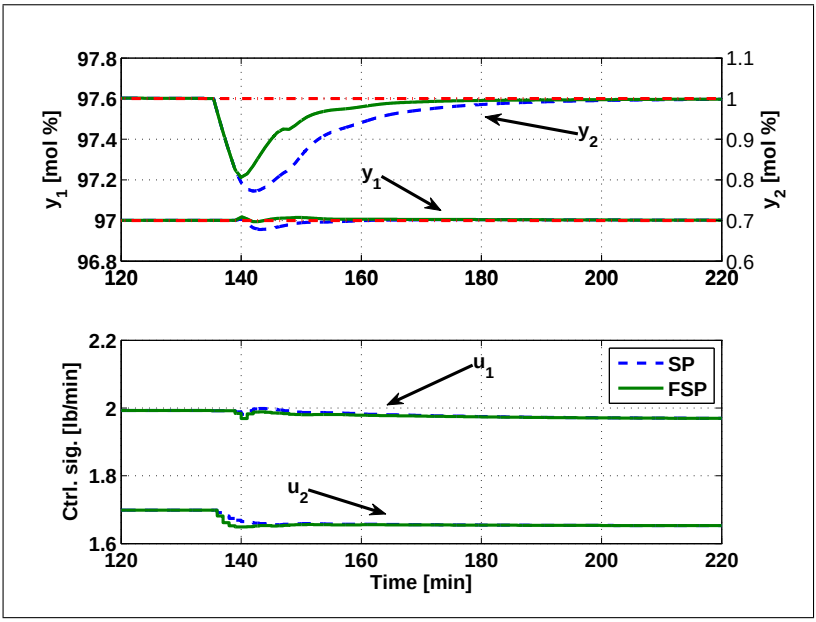

Figure 12 – Nominal disturbance rejection for the MIMO case study.

nal set-point tracking) and Figure 12 (nominal disturbance rejection), and it is shown how the use of the proposed MIMO-FSP can improve the disturbance rejection response. In the nominal case the set-point tracking responses of MIMO-SP and MIMO-FSP are the same, but as MIMO-FSP is tuned to avoid pole-zero cancellations, the disturbance rejection response can be faster than the open-loop dynamics of $\mathbf{P}(z)$.

It is important to take into account the trade-off between the desired disturbance rejection dynamic and the robustness to modelling errors. If an additive bound of $\pm 10\%$ of the static gain is considered around the nominal response with

$$\mathbf{W_1} = \mathbf{I} \quad \text{and} \quad \mathbf{W_2} = \begin{bmatrix} 1.28 & 1.89 \\ 0.66 & 1.94 \end{bmatrix},$$

the value of $1/\left(\overline{\sigma}(\mathbf{M_0})\overline{\sigma}(\mathbf{W}_1)\overline{\sigma}(\mathbf{W}_2)\right)$ is plotted in Figure 13. As can be seen in Figure 13, the condition for the filter $\mathbf{F_{r1}}(z)$ does not assure robust stability and the same can be said about the regular MIMO-SP. In order to guarantee robustness to modelling errors, a simple low-pass

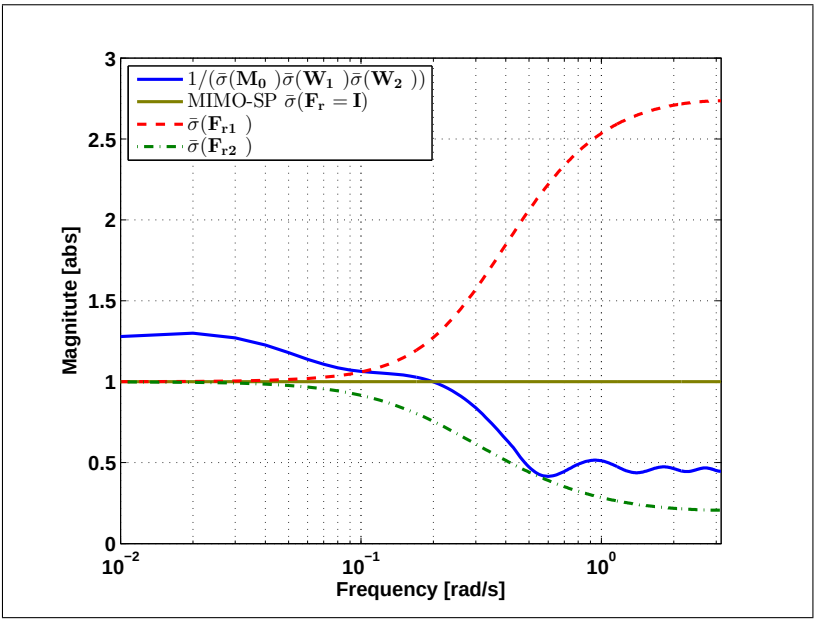

Figure 13 – Robustness condition for the MIMO example.

diagonal filter is used[6]:

$$\mathbf{F_{r2}}(z) = \frac{0.2(z-0.4)}{0.6(z-0.8)}\mathbf{I}.$$

The simulation results with the modelling errors are presented in Figure 14 (note that the magnitude is bigger at low frequencies because of the nature of the uncertainties). Notice that the response with $\mathbf{F_{r1}}(z)$ degenerates[7] because of the model uncertainties, which does not happen with $\mathbf{F_{r2}}(z)$. Obviously, with these uncertainties, it is necessary to prioritize robustness over disturbance rejection so as to maintain the system stability.

[6]Note that as the plant is stable no additional conditions have to be imposed.

[7]From Figure 13, it may seem contradictory that the closed-loop system with $\mathbf{F_{r1}}(z)$ does not unstabilize, but it must be noted that the robustness condition is only sufficient, and not necessary.

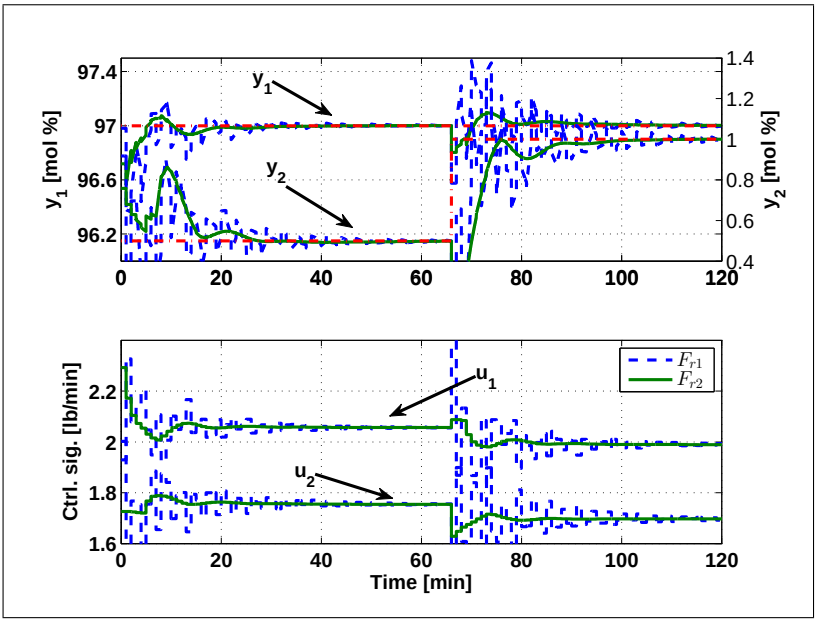

Figure 14 – Set-point tracking for the MIMO case with model mismatch.

2.5 FINAL COMMENTS

This chapter presented the Filtered Smith Predictor Dead-Time Compensation structure which can be used to improve robustness and disturbance rejection of stable and unstable linear processes. The FSP will be used throughout this thesis for the analysis and improvement of a predictive controller algorithm (Chapter 4) and it will be extended to the nonlinear case (Chapters 5 and 6).

3 MODEL PREDICTIVE CONTROL

This chapter will make a brief introduction to Model Predictive Control (MPC) and its basic concepts, with emphasis on the algorithms Dynamic Matrix Control (DMC) (Section 3.2) and Practical Nonlinear Model Predictive Control (PNMPC) (Section 3.3.4), given that they will be used in this thesis.

3.1 MPC OVERVIEW

The term model predictive control does not designate a specific control strategy but a very ample range of control methods that make explicit use of a model of the process to obtain the control signal by minimizing an objective function. The main ideas of the predictive control family are [2]:

- explicit use of a model to predict the process output at future time instants (horizon);
- calculation of a control sequence minimizing an objective function;
- receding strategy, so that at each instant the horizon is displaced towards the future, which involves the application of the first control signal of the sequence calculated at each step.

The differences between the various MPC algorithms are the model used to represent the process, the disturbances and the cost function to be minimized. Several applications of predictive control have been reported in the literature, not only in the process industry but also in specific research applications [64–68]. These applications show the capacity of MPC to achieve highly efficient control systems able to operate over long periods of time with hardly any intervention.

MPC presents a series of advantages over other methods, amongst which the following stand out [3]:

- it can be used to control a great variety of processes, ranging from those with relatively simple dynamics to more complex ones, including systems with long dead time, nonminimum phase and unstable ones;
- the multivariable case can easily be dealt with;

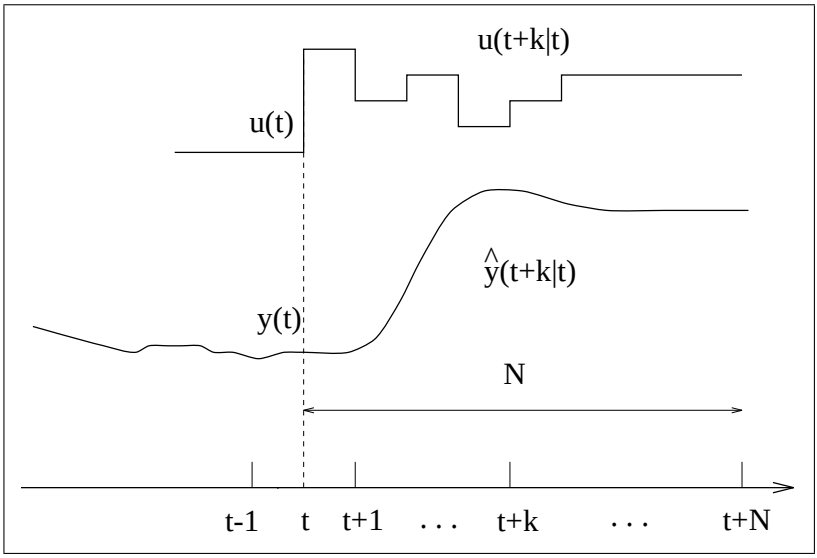

Figure 15 – Basic MPC concepts [2]

- it intrinsically compensates for dead times;
- it introduces feedforward control in a natural way to compensate for measurable disturbances;
- its extension to the treatment of constraints is conceptually simple and these can be systematically included during the design process.

In practice, MPC has proved to be a reasonable strategy for industrial control, and several reports indicate that it is the most used advanced control technology in industry [69].

3.1.1 MPC Strategy

The methodology of all the controllers belonging to the MPC family is characterized by the following strategy (see Figure 15) [3]:

1. The future outputs for a defined horizon N, called the prediction horizon, are predicted at each instant t using the process model.

These predicted outputs $y(t+k|t)$ [1] for $k = 1 \dots N$ depend on the known values up to instant t (past inputs and outputs) and on the future control signals $u(t+k|t)$, $k = 0 \dots N-1$, which are to be sent to the system and to be calculated.

2. The set of future control signals is calculated by optimizing a determined criterion in order to keep the process as close as possible to the reference trajectory $y_r(t+k)$ (which can be the set-point itself or a close approximation of it). This criterion usually takes the form of a quadratic function of the errors between the predicted output signal and the predicted reference trajectory. The control effort is included in the objective function in most cases. An explicit solution can be obtained if the criterion is quadratic, the model is linear and there are no constraints[2], otherwise an iterative optimization method has to be used.

3. The control signal $u(t|t)$ is sent to the process whilst the next control signals calculated are neglected, because, at the next sampling instant, $y(t+1)$ will already be known, and step 1 will be repeated with this new value and all the sequences will be brought up to date. Thus, $u(t+1|t+1)$ is calculated (which in principle will be different to $u(t+1|t)$ because of the new information available) using the receding horizon concept.

The basic structure shown in Figure 16 is used in order to implement this strategy. A model is used to predict the future plant outputs, based on past and current values and on the proposed optimal future control actions. These actions are calculated by the optimizer taking into account the cost function (where the future tracking error is considered) as well as the constraints.

3.1.2 MPC Elements

All the MPC algorithms possess common elements and different options can be chosen for each one of these elements, which gives rise to different algorithms. These elements are

- the prediction model,

[1] The notation indicates the predicted value of the variable at the time instant $t+k$ calculated at instant t.

[2] If there are constraints, an explicit solution can be found, but it is not linear. In this case, the solution is piecewise affine [70, 71].

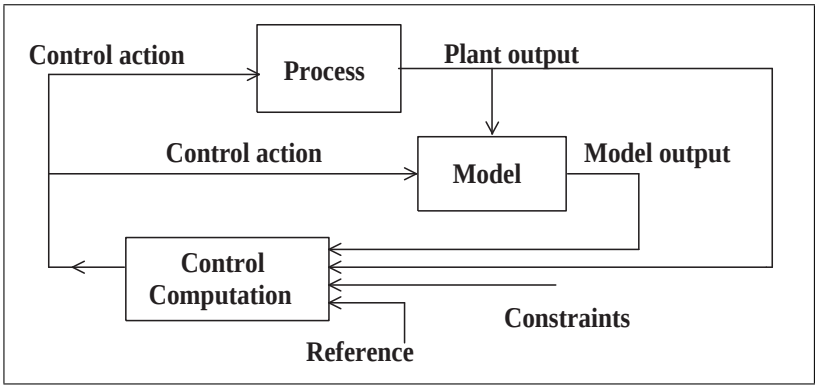

Figure 16 – Basic structure of MPC [3].

- the objective function, and
- the procedure to obtain the control law.

3.1.2.1 The Prediction Model

The process model plays a decisive role in the controller because it is necessary to calculate the predicted output at future instants $y(t+k|t)$. The model can be divided into two parts: the actual process model and the disturbances model. Both parts are needed for the prediction.

Process Model
Various discrete models can be used to represent the process behaviour, but two stand out, the step response and the transfer-function models.

The step response model is used by the DMC algorithm and its variants, and the model equation is given by

$$y(t) = \sum_{i=1}^{\infty} g_i \Delta u(t-i), \tag{3.1}$$

where the coefficients g_i are the sampled output values for the step input and $\Delta u(t) = u(t) - u(t-1)$. This model is widely accepted in industry because it is very intuitive and clearly reflects the influence of each manipulated variable on a determined output. The process of identification is also simplified while still maintaining the ability

to represent complex dynamics such as nonminimum phase and dead time. Although the summation present in the model is infinite, it can be truncated as will be seen later on. One of its main disadvantages is the impossibility of representing unstable processes.

In the transfer function model, the output is given by

$$A(z)y(t) = B(z)u(t-d-1),$$

where d is the dead time, and

$$\begin{aligned} A(z) &= 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_{na} z^{-na}, \\ B(z) &= b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_{nb} z^{-nb}. \end{aligned}$$

This representation is also valid for unstable processes and has the advantage that it only needs a few parameters, although a priori knowledge of the process is fundamental in the case of model identification, especially of the order of the A and B polynomials.

Disturbance Model
Several models can be used to describe the disturbances, that is, the differences between the measured output and the one calculated by the process model. In practice, step or ramp models are usually used and are also considered as particular cases of the disturbance model. When nondeterministic disturbances are considered, such as random changes occurring at random instants, the auto-regressive and integrated moving average (ARIMA) model is widely used. This is given by

$$\eta(t) = \frac{C(z)e(t)}{D(z)\Delta(z)},$$

where $\Delta(z) = 1 - z^{-1}$, $e(t)$ is a white noise of zero mean and the polynomials $C(z)$ and $D(z)$ are used to describe the stochastic characteristics of $\eta(t)$.

3.1.2.2 Free and Forced Response

A typical characteristic of most linear MPCs is the use of *free* and *forced* response concepts. The idea is to express the control sequence as the addition of two signals:

$$u(t) = u_f(t) + u_c(t)$$

where $u_f(t)$ corresponds to the past inputs and is kept constant and equal to the last value of the manipulated variable in future time instants. That is,

$$\begin{aligned} u_f(t-k) &= u(t-k), && \text{for } k = 1, 2, \dots \\ u_f(t+k) &= u(t-1), && \text{for } k = 0, 1, 2, \dots \end{aligned}$$

The signal $u_c(t)$ is made equal to zero in the past and equal to the next control moves in the future. That is,

$$\begin{aligned} u_c(t-k) &= 0, && \text{for } k = 1, 2, \dots \\ u_c(t+k) &= u(t+k) - u(t-1), && \text{for } k = 0, 1, 2, \dots \end{aligned}$$

The prediction of the output sequence is separated into two parts, the *free* response, which corresponds to the prediction of the output when the process manipulated variable is made equal to $u_f(t)$, and the *forced* response, which corresponds to the prediction of the process output when the control sequence is made equal to $u_c(t)$. The *free* response corresponds to the evolution of the process due to its present state, while the forced response is due to the future control moves [3].

3.1.2.3 The Objective Function

The various MPC algorithms propose different cost functions for obtaining the control law. The general aim is that the future predicted output $y(t+k|t)$ on the considered horizon should follow a determined reference signal w and, at the same time, the control effort Δu necessary for doing so should be penalized.

The most common expression for such an objective function, in the SISO case, is

$$J = \sum_{k=N_1}^{N_2} \delta(j)[y(t+k|t) - y_r(t+k)]^2 + \sum_{k=1}^{N_u} \lambda(j)[\Delta u(t+k-1)]^2, \tag{3.2}$$

which is commonly used with the GPC and DMC algorithms.

The tuning parameters are the following [3]:

- Prediction horizons: N_1 and N_2 define the minimum and maximum cost horizons, respectively. They mark the limits of the

instants in which it is desirable for the output to follow the reference.

- Control horizon N_u: indicates in what future instants the controller is allowed to change the input variable. A smaller horizon usually results in a more aggressive closed-loop behaviour.
- Input (λ) and output (δ) weights: they define how the future control increments and error should be penalized in the cost function. They are used to indicate what variable should be prioritized, i.e., if δ is high in relation to λ, it means that the minimization of the cost function will result in a smaller future error, while, if the contrary is true, smaller control increments will be obtained.

An interesting aspect is the consideration of the future reference trajectory. Usually the future evolution of the reference is not known, thus $y_r(t+k) = y_r(t), \forall k > 0$ is used. In some cases, however, for instance in robotics, servos or batch processes, the future reference is known. This information can then be used in the cost function to improve the reference tracking capabilities of the controller [3].

Another advantage of MPC algorithms is that they can explicitly consider the constraints of the process, which are normally defined as bounds in the amplitude and in the slew rate of the control signal, and limits in the output:

$$\begin{aligned} u_{\min} \leq \quad & u(t+k) \quad \leq u_{\max} \qquad \forall k \geq 0, \\ du_{\min} \leq \quad & \Delta u(t+k) \quad \leq du_{\max} \qquad \forall k \geq 0, \\ y_{\min} \leq \quad & y(t+k) \quad \leq y_{\max} \qquad \forall k \geq 0. \end{aligned}$$

3.1.2.4 Obtaining the Control Law

In order to obtain the values $u(t+k|t)$, it is necessary to minimize the function J of Eq. (3.2). To do this the values of the predicted outputs $y(t+k|t)$ are calculated as a function of the past values of the inputs and outputs, and of the future control signals, making use of the model chosen, and then the predicted values are substituted in the cost function, thus resulting in an expression whose minimization leads to the desired values. An analytical solution can be obtained for the quadratic criterion if the model is linear and there are no constraints, otherwise an iterative method of optimization should be used.

There are several MPC algorithms that use the ideas presented in the previous sections. Roughly they can be classified in two main

groups. In the first one we find algorithms derived in industry, like DMC [13] and model algorithm control (MAC) [68]. In these controllers the prediction is based on step or impulse response models of the plant and the disturbances are considered as the difference between the real and predicted outputs [72]. The other class of MPC has mainly been derived in academia based on the ideas of adaptive control, and the algorithms were initially not related to the previous group [73]. In this second group of algorithms, in which GPC [74] and extended prediction self-adaptive control (EPSAC) [75] are included, controlled autoregressive integrated moving average (CARIMA) models are used to represent the plant and disturbances and the predictions of the output are computed using optimal predictors. As representatives of these two groups DMC and GPC are, perhaps, the most popular predictive algorithms [2, 3].

3.2 DYNAMIC MATRIX CONTROL

The DMC strategy uses the cost function introduced earlier in Eq. (3.2), and the predictions of the output of the plant are computed using a step-response model

$$y(t+k|t) = \sum_{i=1}^{\infty} g_i \Delta u(t+k-i) + \eta(t+k|t).$$

where g_i is the ith coefficient of the step response.

The disturbance prediction $\eta(t+k|t)$ is considered constant along the horizon and equal to the difference between the process and model output

$$\eta(t+k|t) = \eta(t|t) = y(t) - y(t|t).$$

Using these two expressions and separating the future control actions gives

$$\begin{aligned} y(t+k|t) = &\sum_{i=1}^{k} g_i \Delta u(t+k-i) + \sum_{i=k+1}^{\infty} g_i \Delta u(t+k-i) + y(t) \\ &- \sum_{i=1}^{\infty} g_i \Delta u(t-i) \\ = &\sum_{i=1}^{k} g_i \Delta u(t+k-i) + f(t+k), \end{aligned} \tag{3.3}$$

where $f(t+k)$ is the free response of the system, that is, the part of the response that does not depend on the future control actions and is given by

$$f(t+k) = \sum_{i=1}^{\infty} (g_{k+i} - g_i)\Delta u(t-i). \tag{3.4}$$

If the process is asymptotically stable, the coefficients g_i of the step response tend to a constant value after M sampling periods, so it can be considered that

$$g_{k+i} - g_i \approx 0, \qquad i > M,$$

with the parameter M being the model horizon, which is usually defined during the identification of the process model. Thus,

$$y(t+k|t) = \sum_{i=1}^{k} g_i \Delta u(t+k-i) + y(t) + \sum_{i=1}^{M} (g_{k+i} - g_i)\Delta u(t-i). \tag{3.5}$$

Notice that if the process is not asymptotically stable, then M does not exist and $f(t+k)$ cannot be computed (although a generalization exists in the case of the instability being produced by pure integrators [72]).

Thus, using a prediction horizon and a control horizon, the minimization of J is accomplished using the predicted values.

3.2.1 Computing the Predictions

Using Eq. (3.5), and assuming without loss of generality, that the initial and final prediction horizons are, respectively, $N_1 = 1$ and $N_2 = N$, it is possible to write the prediction in matrix form

$$\hat{\mathbf{y}} = \mathbf{G}\boldsymbol{\Delta}\mathbf{u}(t) + \mathbf{H}\boldsymbol{\Delta}\mathbf{u}(t-1) + \mathbf{1}y(t) \tag{3.6}$$

where

$$\begin{aligned}
\hat{\mathbf{y}} &= [y(t+1|t), \ldots, y(t+N|t)]^T \\
\boldsymbol{\Delta}\mathbf{u}(t) &= [\Delta u(t), \Delta u(t+1), \ldots, \Delta u(t+N_u-1)]^T \\
\boldsymbol{\Delta}\mathbf{u}(t-1) &= [\Delta u(t-1), \Delta u(t-2), \ldots, \Delta u(t-M)]^T,
\end{aligned}$$

and $\mathbf{1}$ is a vector with dimension N whose elements are all ones. Matrices $\mathbf{G}$ and $\mathbf{H}$ have dimension $N \times N_u$ and $N \times M$, respectively, and are given by

$$\mathbf{G} = \begin{bmatrix} g_1 & 0 & \dots & 0 \\ g_2 & g_1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_{N_u} & g_{N_u-1} & \cdots & g_1 \\ \vdots & \vdots & \dots & \vdots \\ g_N & g_{N-1} & \cdots & g_{N-N_u+1} \end{bmatrix},$$

$$\mathbf{H} = \begin{bmatrix} (g_2 - g_1) & (g_3 - g_2) & \dots & (g_{M+1} - g_M) \\ (g_3 - g_1) & (g_4 - g_2) & \dots & (g_{M+2} - g_M) \\ \vdots & \vdots & \ddots & \vdots \\ (g_{N+1} - g_1) & (g_{N+2} - g_2) & \dots & (g_{M+N} - g_M) \end{bmatrix}.$$

The free response of the system is given by $\mathbf{f} = \mathbf{H}\ \boldsymbol{\Delta}\mathbf{u}(t-1) + \mathbf{1}y(t)$. Thus, the obtained predictions are used in the minimization of J.

3.2.2 Minimization of J

To obtain the vector of future control increments $\boldsymbol{\Delta}\mathbf{u}(t)$ it is necessary to minimize the cost function J, which can be rewritten in matrix form

$$J = (\hat{\mathbf{y}} - \mathbf{y_r})^T \mathbf{Q_y} (\hat{\mathbf{y}} - \mathbf{y_r}) + \boldsymbol{\Delta}\mathbf{u}(t)^T \mathbf{Q_u} \boldsymbol{\Delta}\mathbf{u}(t) \tag{3.7}$$

where $\mathbf{Q_u}$ and $\mathbf{Q_y}$ are diagonal matrices whose elements are the input and error weights, respectively, with dimension $N_u \times N_u$ and $N \times N$, and $\mathbf{y_r}$ is the vector of future set-points.

Using Eq. (3.7), and after some rearrangements, the cost function can be rewritten in the standard quadratic problem form

$$J = \frac{1}{2} \boldsymbol{\Delta}\mathbf{u}(t)^T \mathbf{H_{qp}} \boldsymbol{\Delta}\mathbf{u}(t) + \mathbf{b}^T \boldsymbol{\Delta}\mathbf{u}(t) + \mathbf{f_0}$$

where

$$\begin{aligned}\mathbf{H_{qp}} &= 2(\mathbf{G}^T\mathbf{Q_y}\mathbf{G} + \mathbf{Q_u})\\ \mathbf{b}^T &= 2(\mathbf{f} - \mathbf{y_r})^T\mathbf{Q_y}\mathbf{G}\\ \mathbf{f_0} &= (\mathbf{f} - \mathbf{y_r})^T(\mathbf{f} - \mathbf{y_r})\end{aligned}$$

The minimum of J, assuming there are no constraints, can be found by making the gradient of J equal to zero, which leads to:

$$\begin{aligned}\boldsymbol{\Delta}\mathbf{u}(t) = &-\mathbf{H_{qp}}^{-1}\mathbf{b} = (\mathbf{G}^T\mathbf{Q_y}\mathbf{G} + \mathbf{Q_u})^{-1}\mathbf{G}^T\mathbf{Q_y}(\mathbf{y_r} - \mathbf{f})\\ =&\mathbf{K}(\mathbf{y_r} - \mathbf{f}),\end{aligned}$$

where $\mathbf{K} = (\mathbf{G}^T\mathbf{Q_y}\mathbf{G} + \mathbf{Q_u})^{-1}\mathbf{G}^T\mathbf{Q_y}$.

Notice that the control signal that is actually sent to the process is the first element of vector $\boldsymbol{\Delta}\mathbf{u}(t)$, that is given by

$$\Delta u(t) = \mathbf{K_1}(\mathbf{y_r} - \mathbf{f}),$$

where $\mathbf{K_1}$ is the first row of $\mathbf{K}$. This has a clear meaning, if there are no future predicted errors, i.e., if $\mathbf{y_r} - \mathbf{f} = 0$, then there is no control move, since the objective will be fulfilled with the free evolution of the process. However, in the other case, there will be an increment in the control action proportional (with a factor $\mathbf{K}$) to that predicted future error [2].

3.3 NONLINEAR MODEL PREDICTIVE CONTROL

Linear MPC are the most used strategies in practice. However, normally processes exhibit nonlinear behavior, thus, linearized models are used to compute the predictions. To illustrate the relationship between the nonlinear general model and the linearized one, consider the following nonlinear system in state-space form, which is generally the representation used in this case:

$$\begin{cases}\mathbf{x}(t+1) = f(\mathbf{x}(t), \mathbf{u}(t)) & \text{(3.8a)}\\ \mathbf{y}(t) = h(\mathbf{x}(t)) & \text{(3.8b)}\end{cases}$$

where $f(\mathbf{x}, \mathbf{u}) : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ and $h(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}^{n_y}$ are continuous differentiable functions, and n, n_y and m are the number of states, outputs and inputs, respectively.

From this nonlinear equation, it is possible to derive a linearized

state-space model around a given operating point $\mathbf{x}^*, \mathbf{u}*$

$$\begin{aligned}\mathbf{x}(t+1) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)\\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t)\end{aligned}$$

where

$$\mathbf{A} = \left.\frac{\partial f}{\partial x}\right|_{\mathbf{x}^*,\mathbf{u}*}, \qquad \mathbf{B} = \left.\frac{\partial f}{\partial u}\right|_{\mathbf{x}^*,\mathbf{u}*}, \qquad \mathbf{C} = \left.\frac{\partial h}{\partial x}\right|_{\mathbf{x}^*,\mathbf{u}*}.$$

The eigenvalues of matrix $\mathbf{A}$ indicate the stability of the operating point. Furthermore, the continuous state-space model of Eqs. (3.8a) and (3.8b) can be converted to discrete time easily.

However, when the complexity of the process is high, it is necessary to use strategies that consider the nonlinearities present in the process. It is important to note that the use of MPC with nonlinear processes does not change the MPC philosophy, but the complexity increases. In this section some of these techniques will be briefly described.

3.3.1 Methods with Generic Nonlinear Models

In these cases the cost function is formulated as

$$\begin{aligned}&J = (\hat{\mathbf{y}} - \mathbf{y_r})^T \mathbf{Q_y} (\hat{\mathbf{y}} - \mathbf{y_r}) + \boldsymbol{\Delta}\mathbf{u}(t)^T \mathbf{Q_u} \boldsymbol{\Delta}\mathbf{u}(t)\\ &\text{Subject to:}\\ &\mathbf{x}(k+1) = f(\mathbf{x}(t), \mathbf{u}(\mathbf{t}))\\ &\mathbf{y}(t) = h(\mathbf{x}(t))\end{aligned}$$

where f and h are continuous and differentiable nonlinear functions.

With this formulation, the minimization of the cost function J must be done using nonlinear optimization algorithms, e.g., Sequential Quadratic Programming (SQP). As discussed in [76], these methods have some disadvantages, i.e., the time to obtain the solution can have large variations because of convergence of the algorithm, the presence of local minima, etc.

3.3.2 Methods with Particular Nonlinear Models

When a process can be described with specific models, there are optimization techniques that use the particularities of the model to obtain better performance, hence simplifying the minimization of J. This is the case for Volterra Series model [77] and Hammerstein models [78, 79]. However, it is necessary that the process can be accurately described by these models, otherwise it may not be possible to maintain the robustness of the closed-loop system if the operating point changes.

3.3.3 Methods with Linearized Models

This strategy consists of using a linear MPC which incorporates linearizations in a set of operating points and techniques to cope with modelling errors. Some examples are listed below:

- Multiple Linearized models: in some cases, it is possible to obtain various linearized models in different operating points, i.e., given the nonlinear system described by Eq. (3.8), for each operating point $(\mathbf{x}^{*\mathbf{i}}, \mathbf{u}^{*\mathbf{i}})$ where $\dot{\mathbf{x}} = f(\mathbf{x}^{*\mathbf{i}}, \mathbf{u}^{*\mathbf{i}}) = 0$, there is a linearized model of the form

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A_i}(\mathbf{x} - \mathbf{x}^{*\mathbf{i}}) + \mathbf{B_i}(\mathbf{u} - \mathbf{u}^{*\mathbf{i}}) \\ \mathbf{y} &= h(\mathbf{x}^{*\mathbf{i}}) + \mathbf{C_i}(\overline{\mathbf{x}} - \mathbf{x}^{*\mathbf{i}})\end{aligned}$$

  With $\mathbf{A_i} = \left.\frac{\partial f}{\partial x}\right|_{\mathbf{x}^{*\mathbf{i}}, \mathbf{u}^{*\mathbf{i}}}$, $\mathbf{B_i} = \left.\frac{\partial f}{\partial u}\right|_{\mathbf{x}^{*\mathbf{i}}, \mathbf{u}^{*\mathbf{i}}}$ and $\mathbf{C_i} = \left.\frac{\partial h}{\partial x}\right|_{\mathbf{x}^{*\mathbf{i}}, \mathbf{u}^{*\mathbf{i}}}$. These continuous models can be discretized, obtaining matrices $\mathbf{A_{di}}$, $\mathbf{B_{di}}$ and $\mathbf{C_{di}}$. Then the linearized process' matrices can be represented in the polytopic form

$$[\mathbf{A_{di}}, \mathbf{B_{di}}, \mathbf{C_{di}}] = \sum_{i=1}^{N} \alpha_i [\mathbf{A_{di}}, \mathbf{B_{di}}, \mathbf{C_{di}}], \quad \text{where } \sum_{i=1}^{N} \alpha_i = 1,$$

  In general, the value of α_i is not easy to determine. This approach was studied in the context of robust MPC in [80] and adaptive MPC in [81].

- Linearization at each sampling instant: there are nonlinear MPC algorithms that use the concept of free and forced response, i.e.,

they use an approximated nonlinear model of the process to compute the free response and a linearized model to compute the forced response. Hence the nonlinear optimization problem can be transformed in a sequence of linear or quadratic optimization problems [82]. However, this strategy does not guarantee that the result of the optimization is the global optimal value.

3.3.4 Practical Nonlinear Model Predictive Control

The nonlinear MPC algorithm that will be used throughout this thesis was first proposed by Plucênio in [82, 83], and was titled Practical Nonlinear MPC (PNMPC). The objective of this algorithm is to provide a simple and efficient way to apply the concept of model predictive control to nonlinear processes while avoiding a complex nonlinear optimization problem.

The Linear MPC algorithms use the concept of free and forced response, i.e., the prediction vector $\hat{\mathbf{y}}$ can be written as

$$\hat{\mathbf{y}} = \mathbf{G}\boldsymbol{\Delta}\mathbf{u}(t) + \mathbf{f}.$$

Since the system in consideration is nonlinear, the superposition principle cannot be applied and it is not possible to obtain this separation between free and forced responses. However, the PNMPC algorithm ingeniously manages to separate this responses linearizing the nonlinear equation at each time sample as will be seen in Section 3.3.4.1.

3.3.4.1 Obtaining the Forced Response

For the sake of presentation simplicity, consider the following first order SISO discrete nonlinear system:

$$y(t+1) = f(y(t), u(t)) + w(t), \tag{3.9}$$

where y is the measured output, u is the input, w is a disturbance that can describe an external perturbation, model mismatch or noise, and $f : \mathbb{R}\times\mathbb{R} \to \mathbb{R}$. Notice that the output value at $t+1$ only depends on the input and output at time t. This assumption may seem very restrictive but the results that will be demonstrated can easily be extended for a more general nonlinear process.

To proceed, it is necessary to obtain the open-loop prediction vector $\overline{\mathbf{y}}$ (predictions without taking into account disturbances), but, before that, consider that $u(t) = u(t-1) + \Delta u(t)$, hence, the open-loop prediction at the time instant $t+1$, given the information at time t, can be written as

$$\overline{y}(t+1|t) = f_0(y(t), u(t-1), \Delta u(t)),$$

where $f_0(.) = f(.)$ (the necessity of this notation will become clear soon).

The future predictions can be computed recursively through the use of Eq. (3.9). For instance, the prediction at the time instant $t+2$ is

$$\begin{aligned}\overline{y}(t+2|t) &= f(\overline{y}(t+1|t), u(t+1)) \\ &= f(f_0(y(t), u(t-1), \Delta u(t)), u(t+1))\end{aligned}$$

and, given that $u(t+1) = u(t-1) + \Delta u(t) + \Delta u(t+1)$,

$$\overline{y}(t+2|t) = f_1(y(t), u(t-1), \Delta u(t), \Delta u(t+1)),$$

with $f_1(.)$ being the result of a function composition. By inspection, it is easy to see that the vector of future predictions is

$$\begin{bmatrix} \overline{y}(t+1|t) \\ \overline{y}(t+2|t) \\ \vdots \\ \overline{y}(t+N|t) \end{bmatrix} = \begin{bmatrix} f(y(t), u(t-1), \Delta u(t)) \\ f_1(y(t), u(t-1), \Delta u(t), \Delta u(t+1)) \\ \vdots \\ f_{N-1}(y(t), u(t-1), \Delta u(t), \ldots, \Delta u(t+N-1)) \end{bmatrix},$$

where N is the prediction horizon and, to simplify, it is considered that the control horizon is $N_u = N$. This last equation can be rearranged as

$$\overline{\mathbf{y}} = \mathbf{F}(y(t), u(t-1), \mathbf{\Delta u}(t)) \tag{3.10}$$

where $\mathbf{\Delta u}(t) = [\Delta u(t), \Delta u(t+1), \ldots, \Delta u(t+N-1)]^T$.

Now, consider that, for a given nonlinear function, it is possible to linearize it around an operating point $\mathbf{x}^*$ using a first order Taylor series approximation, i.e.,

$$\mathbf{g}(\mathbf{x}^* + \mathbf{\Delta x}) \cong \mathbf{g}(\mathbf{x}^*) + \left.\frac{\partial \mathbf{g}}{\partial \mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}^*} \mathbf{\Delta x}.$$

Applying this concept to Eq. (3.10),

$$\overline{\mathbf{y}} \cong \mathbf{F}(\mathbf{x}^*) + \left.\frac{\partial \mathbf{F}}{\partial \boldsymbol{\Delta}\mathbf{u}(t)}\right|_{\mathbf{x}=\mathbf{x}^*} \boldsymbol{\Delta}\mathbf{u}(t) + \left.\frac{\partial \mathbf{F}}{\partial y(t)}\right|_{\mathbf{x}=\mathbf{x}^*} \Delta y(t) + \left.\frac{\partial \mathbf{F}}{\partial u(t-1)}\right|_{\mathbf{x}=\mathbf{x}^*} \Delta u(t-1), \tag{3.11}$$

where $\mathbf{x} = [y(t), u(t-1), \boldsymbol{\Delta}\mathbf{u}(t)]$ and the operating point is $\mathbf{x}^* = [y(t), u(t-1), \mathbf{0}]$, with $\mathbf{0}$ being a vector with appropriate dimension whose elements are all zeros. The two last terms of the right side of Eq. (3.11) are null because $y(t)$ and $u(t-1)$ are constants at the sampling instant t, hence, $\Delta y(t) = \Delta u(t-1) = 0$. This reduces the prediction equation to

$$\overline{\mathbf{y}} \cong \mathbf{F}(\mathbf{x}^*) + \left.\frac{\partial \mathbf{F}}{\partial \boldsymbol{\Delta}\mathbf{u}(t)}\right|_{\mathbf{x}=\mathbf{x}^*} \boldsymbol{\Delta}\mathbf{u}(t).$$

Let's analyze each term separately. The first term is the value of $\mathbf{F}(\mathbf{x})$ for the operating point, which is $\mathbf{x}^* = [y(t), u(t-1), \boldsymbol{\Delta}\mathbf{u} = \mathbf{0}]$. Note that all future control increments are equal to zero, hence, this is exactly the free response of the nonlinear process, i.e., the natural behaviour of the system if the control action is maintained at its current value.

The second term represents how the function $\mathbf{F}(\mathbf{x})$ varies around the operating point when the control increments at different future instants varies, which is the forced response of the nonlinear process. Therefore, the PNMPC algorithm computes the prediction of a nonlinear process in the following way

$$\overline{\mathbf{y}} = \mathbf{G}_{\mathbf{PNMPC}} \boldsymbol{\Delta}\mathbf{u} + \mathbf{f},$$

where $\mathbf{G}_{\mathbf{PNMPC}}$ is the Jacobian matrix of $\mathbf{F}(\mathbf{x})$ in relation to $\boldsymbol{\Delta}\mathbf{u}(t)$, i.e.,

$$\mathbf{G}_{\mathbf{PNMPC}} = \frac{\partial \mathbf{F}}{\partial \boldsymbol{\Delta}\mathbf{u}(t)} = \begin{bmatrix} \frac{\partial y(t+1|t)}{\partial \Delta u(t)} & 0 & \cdots & 0 \\ \frac{\partial y(t+2|t)}{\partial \Delta u(t)} & \frac{\partial y(t+2|t)}{\partial \Delta u(t+1)} & \cdots & 0 \\ \vdots & \cdots & \ddots & \vdots \\ \frac{\partial y(t+N|t)}{\partial \Delta u(t)} & \frac{\partial y(t+N|t)}{\partial \Delta u(t+1)} & \cdots & \frac{\partial y(t+N|t)}{\partial \Delta u(t+N-1)} \end{bmatrix}$$

Note the lower triangular characteristics of $\mathbf{G}_{\mathbf{PNMPC}}$, which is rather intuitive. Since the system is causal, the prediction at time $t+k$

depends only on control increments up to time $t + i - 1$.

To compute the Jacobian matrix $\mathbf{G_{PNMPC}}$ analytically can be extremely complex depending of the nonlinearities present in the system equation, thus, another solution is required. Using the concept of limits, the partial derivative of a function is given by

$$\frac{\partial g(x)}{\partial x} = \lim_{\Delta x \to 0} \frac{g(x + \Delta x) - g(x)}{\Delta x}.$$

This can be used to compute numerically the Jacobian of $\mathbf{f}(\mathbf{x})$:

1. Compute the free response of the system $\overline{\mathbf{y}}_\mathbf{0} = \mathbf{f}(\overline{\mathbf{x}})$.

2. Compute the first column of $\mathbf{G_{PNMPC}}$. Make

   $$\mathbf{\Delta u}(t) = [\epsilon, 0, \ldots, 0]^T,$$

   where ϵ is a small value, e.g., $u(t-1)/1000$, and calculate $\overline{\mathbf{y}}_\mathbf{1} = \mathbf{F}(y(t), u(t-1), \mathbf{\Delta u}(t))$. Then, the first column of $\mathbf{G_{PNMPC}}$ is given by:

   $$\frac{\overline{\mathbf{y}}_\mathbf{1} - \overline{\mathbf{y}}_\mathbf{0}}{\epsilon}.$$

3. Compute the second column of $\mathbf{G_{PNMPC}}$. Make

   $$\mathbf{\Delta u}(t) = [0, \epsilon, 0, \ldots, 0]^T,$$

   and calculate $\overline{\mathbf{y}}_\mathbf{2} = \mathbf{F}(y(t), u(t-1), \mathbf{\Delta u}(t))$. Then, the second column of $\mathbf{G_{PNMPC}}$ is given by:

   $$\frac{\overline{\mathbf{y}}_\mathbf{2} - \overline{\mathbf{y}}_\mathbf{0}}{\epsilon}.$$

4. Repeat the same procedure for the other columns of $\mathbf{G_{PNMPC}}$ until the matrix is complete.

3.3.4.2 Obtaining the Free Response

The free response of the model computed in Section 3.3.4.1 is only used to compute the matrix $\mathbf{G_{PNMPC}}$ because it does not take into account the presence of disturbances, hence, the resulting controller will not be able to reject these signals. To compute the corrected free response, the PNMPC models the disturbance as an integrated white

noise, which is sufficient to describe constant process disturbances,

$$y(t+1) = f(y(t), u(t)) + \frac{e(t)}{\Delta},$$

where $e(t)$ is a white noise and $\Delta = 1 - q^{-1}$ is a polynomial in the delay operator q, i.e., $q^{-1} f(x(t), u(t)) = f(x(t-1), u(t-1))$. Multiplying both sides by Δ,

$$\Delta y(t+1) = \Delta f(y(t), u(t)) + e(t).$$

Since the expected value of $e(t) = 0$, $\forall t$, the optimal prediction at $t+1$ is

$$\begin{aligned} \Delta y(t+1|t) &= \Delta f(y(t), u(t)) \\ y(t+1|t) - y(t|t) &= f(y(t), u(t)) - f(y(t-1), u(t-1)), \end{aligned}$$

and given that the prediction $y(t|t) = y(t)$, i.e., it is the measured output value,

$$y(t+1|t) = f(y(t), u(t)) + y(t) - f(y(t-1), u(t-1)).$$

Notice that $y(t) - f(y(t-1), u(t-1))$ is the prediction error, i.e., the difference between the output at time t minus the prediction $y(t|t-1)$, or, $y(t) - y(t|t-1)$.

The same procedure can be used to compute the prediction at time $t+2$

$$\begin{aligned} \Delta y(t+2|t) &= \Delta f(y(t+1|t), u(t+1)) \\ y(t+2|t) - y(t+1|t) &= f(y(t+1|t), u(t+1)) - f(y(t|t), u(t)), \end{aligned}$$

substituting $y(t+1|t)$ in this last equation,

$$y(t+2|t) = f(y(t+1|t), u(t+1)) + y(t) - f(y(t-1), u(t-1)).$$

By inspection, it can be seen that the predictions can be computed recursively with the addition of the prediction error:

$$\begin{aligned} y(t+k|t) =& f(y(t+k-1|t), u(t+k-1)) \\ &+ y(t) - f(y(t-1), u(t-1)). \end{aligned} \tag{3.12}$$

3.3.5 PNMPC Algorithm

Thus, to apply the PNMPC algorithm the following steps are necessary:

1. read the current output of the system $y(t)$;
2. compute $\mathbf{G_{PNMPC}}$, through the method described in section 3.3.4.1;
3. compute the corrected free response $\mathbf{f}$ of the system, using Eq. (3.12) recursively;
4. minimize the quadratic cost function J, Eq. (3.7), and obtain the current control increments;
5. apply the calculated control action in the nonlinear process;
6. wait one sample time, and repeat.

Thus, if compared to Linear MPC, the increased complexity with the use of the PNMPC technique appears only in the computation of the matrix $\mathbf{G_{PNMPC}}$, which must be done at each time sample, and the use of a nonlinear model to compute the free response. However, the cost function remains a quadratic function that must be solved only once at each time sample, which is far less complex than the other nonlinear MPC techniques which require the solution of a nonlinear programming or an SQP problem.

3.4 FINAL COMMENTS

This chapter reviewed the basic concepts of linear and nonlinear Model Predictive Controllers. The concepts of free and forced response were introduced, and the MPC tuning parameters (prediction and control horizons, weights) were presented. In the next chapter a modification for the DMC algorithm which uses the FSP concepts to improve control performance for dead-time processes will be proposed. And, in Chapter 5, the PNMPC will be used to evaluate the proposed Nonlinear Filtered Smith Predictor.

4 FILTERED DYNAMIC MATRIX CONTROL

As representatives of the MPC algorithms, DMC and GPC are, likely, the most popular predictive algorithms, and in the last two decades several papers have analyzed their performance and robustness [72, 74, 84–88]. However, in spite of the fact that these controllers can cope with processes with dead time, most of the above papers do not pay sufficient attention to the robustness problems caused by errors in the estimation of dead time. As has been pointed out in Chapter 2, dead time uncertainties are one of the most characteristic types of high-frequency unmodelled dynamics and have a dangerous influence on the closed-loop stability. For the particular case of GPC, in [3, 89], the robustness of the algorithm for dead time processes was analyzed through an FSP interpretation, and it was demonstrated that substituting the implicit predictor structure of the GPC by an FSP could, with appropriate tuning, lead to improved robustness. In [3], it was suggested that the same modification could be done to the DMC, but it was not demonstrated how this would be achieved. Thus, in this chapter a special analysis of the effect of dead time on the control formulation of the DMC is performed based on the FSP structure and, in section 4.2, a modification to the original algorithm, called Filtered DMC (FDMC), that allows an improved robustness and disturbance rejection, is proposed. In section 4.3 the implementation of the new algorithm is discussed. Finally, in section 4.4 simulation results are presented to illustrate the advantages of FDMC over the traditional DMC. The results of this chapter were published in [90].

4.1 DTC INTERPRETATION OF DMC

The DTC structure used to analyze the DMC is the Filtered Smith Predictor (FSP) [3], which comprises a primary controller $\mathbf{C}(z)$ and a predictor structure, that are presented in Figure 17. Considering a MIMO system with m inputs, n outputs and p output disturbances, $\mathbf{n}(t)$ and $\mathbf{q}(t)$ are vectors that represent, respectively, output and input disturbances, $\mathbf{F_r}(z)$ is an $n \times n$ matrix transfer function which will be tuned to modify the robustness and disturbance rejection properties of the system, $\mathbf{r(t)}$ represents the set-points, $\mathbf{F}(z)$ is an $n \times n$ matrix transfer function that defines the reference filters, $\mathbf{C}(z)$ is an $m \times n$ controller tuned to stabilize the system's nominal fast model $\mathbf{G_n}(z)$

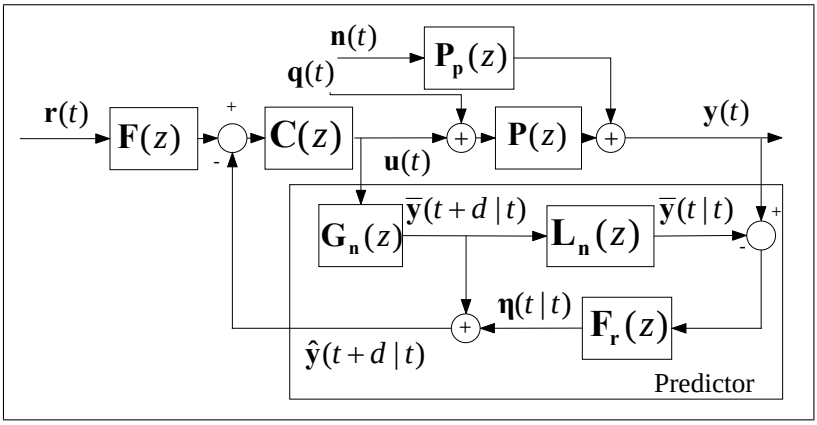

Figure 17 – Structure of the FSP.

(the full nominal system model is given by $\mathbf{P_n}(z) = \mathbf{L_n}(z)\mathbf{G_n}(z)$, in which $\mathbf{L_n}(z)$ is related to the time delay). $\mathbf{P_p}(z)$ represents how $\mathbf{n}(t)$ affects $\mathbf{y}(t)$, and the plant is defined as

$$\mathbf{P}(z) = \begin{bmatrix} G_{11}(z)z^{-d_{11}} & \cdots & G_{1m}(z)z^{-d_{1m}} \\ G_{21}(z)z^{-d_{21}} & \cdots & G_{2m}(z)z^{-d_{2m}} \\ \vdots & \ddots & \vdots \\ G_{n1}(z)z^{-d_{n1}} & \cdots & G_{nm}(z)z^{-d_{nm}} \end{bmatrix}.$$

In this model, $G_{ij}(z)z^{-d_{ij}}$ is the transfer function relating the jth input with the ith output, where $G_{ij}(z)$ is a delay-free transfer function and d_{ij} is the discrete dead time. The effective dead time of each output i is d_i, computed as the minimal delay of the ith row, i.e., $d_i = \min_{j=1\ldots n}(d_{ij})$. Thus, defining $\mathbf{L}(z) = diag\{z^{-d_1}, \ldots, z^{-d_n}\}$ as the MIMO delay of the plant $\mathbf{P}(z)$ and $\mathbf{G}(z)$ as the model without the common dead times (also called fast model), it follows that $\mathbf{P}(z) = \mathbf{L}(z)\mathbf{G}(z)$. It is worth noting that $\mathbf{G}(z)$ may still contain multiple delays [1].

Now it will be shown that the unconstrained DMC algorithm can be interpreted as a DTC controller. DMC uses the step response of the system to compute the predictions of the outputs. For a MIMO system with m inputs and n outputs, the prediction of the ith output

at the future instant $t+k$, given the information at time t, is

$$y_i(t+k|t) \quad = \quad \overline{y}_i(t+k|t) + \eta_i(t+k|t) \tag{4.1}$$

where $\overline{y}_i$ is the open-loop prediction of the ith output of the system, which is given in Eq. (4.2), and $\eta_i(t+k|t)$ is a disturbance that represents the prediction error, or the difference between the measured output $y_i(t)$ and the one calculated by the step-response model [2]. Since the future prediction error is usually not known, DMC considers all future errors equal to the current one, i.e., $\eta_i(t+k|t) = \eta_i(t|t) = y_i(t) - \overline{y}_i(t|t)$. The open-loop prediction is given by

$$\overline{y}_i(t+k|t) = \sum_{j=1}^{m} \left(\sum_{l=1}^{\infty} g_{ij,l} \Delta u_j(t+k-l) \right), \tag{4.2}$$

where $g_{ij,l}$ is the lth step response coefficient of the ith output in relation to the jth input.

4.1.1 Obtaining the Primary Controller

To obtain the primary controller, the first step is to write the predictions of the outputs after the minimum dead time d_i ($y_i(t+d_i+k|t)$ with $k>0$), as a function of the prediction at the time instant $t+d_i$, or $y_i(t+d_i|t)$. Rewriting Eq. (4.1), the following equations are obtained:

$$\begin{aligned} y_i(t+d_i+k|t) \quad = \quad & \sum_{j=1}^{m} \left(\sum_{l=1}^{\infty} g_{ij,d_i+l} \Delta u_j(t+k-l) \right) \\ & + y_i(t) - \overline{y}_i(t|t) \end{aligned} \tag{4.3}$$

and

$$\begin{aligned} y_i(t+d_i|t) \quad = \quad & \sum_{j=1}^{m} \left(\sum_{l=1}^{\infty} g_{ij,d_i+l} \Delta u_j(t-l) \right) \\ & + y_i(t) - \overline{y}_i(t|t). \end{aligned} \tag{4.4}$$

Notice that the coefficients $g_{ij,l}$ for $1 \leq l \leq d_i$ are all zero because of the dead time, thus, they are disconsidered. Isolating $y_i(t)$ in Eq. (4.4) and substituting in Eq. (4.3), it is possible to obtain, after some

rearrangements,

$$\begin{aligned} y_i(t+d_i+k|t) &= \sum_{j=1}^{m}\left(\sum_{l=1}^{k} g_{ij,d_i+l}\Delta u_j(t+k-l)\right. \\ &\quad \left. -\sum_{l=1}^{\infty} a_{ij,l}\Delta u_j(t-l)\right) \\ &\quad +y_i(t+d_i|t), \end{aligned} \tag{4.5}$$

where $a_{ij,l} = g_{ij,d_i+k+l} - g_{ij,d_i+l}$.

In Eq. (4.5), the future control increments and past ones were split in different summations. Notice, however, that the summation of the past increments has infinite terms. Nevertheless, for stable systems, those summations can be truncated, because

$$a_{ij,l} = g_{ij,d_i+k+l} - g_{ij,d_i+l} \cong 0,$$

after M terms, where M is a sufficiently large number [2].

With this last equation, the primary controller of the FSP structure can be obtained. Rewriting Eq. (4.5) in matrix form, it follows that

$$\hat{\mathbf{Y}} = \mathbf{G}\boldsymbol{\Delta}\mathbf{U} + \mathbf{H}\boldsymbol{\Delta}\mathbf{u}(t-1) + \mathbf{1}\mathbf{y}(t+d|t) \tag{4.6}$$

where $\mathbf{G}$ and $\mathbf{H}$ are matrices with dimensions $\sum_{i=1}^{n} N_{yi} \times \sum_{i=1}^{m} N_{ui}$ and $\sum_{i=1}^{n} N_{yi} \times mM$, respectively. The prediction and control horizons of the ith output and ith input are, respectively, N_{yi} and N_{ui}. The number of step coefficients obtained with the step response of the system for each input/output pair is M[1]. $\mathbf{1}$ is a block diagonal matrix composed of n column vectors with dimension N_{yi}, with all entries equal to one. The future predictions are given by the vector $\hat{\mathbf{Y}} = [\hat{\mathbf{y}}_\mathbf{1}, \ldots, \hat{\mathbf{y}}_\mathbf{n}]^T$, where $\hat{\mathbf{y}}_\mathbf{i} = [y_i(t+d_i+1|t), \ldots, y_i(t+d_i+N_{yi}|t)]^T$, the future control increments vector is $\boldsymbol{\Delta}\mathbf{U} = [\mathbf{u}_\mathbf{1}, \ldots, \mathbf{u}_\mathbf{m}]^T$, where $\mathbf{u}_\mathbf{i} = [\Delta u_i(t), \ldots, \Delta u_i(t+N_{ui}-1)]^T$, and the past control increments vector is $\boldsymbol{\Delta}\mathbf{u}(t-1) = [\boldsymbol{\Delta}\mathbf{u}_\mathbf{1}(t-1), \ldots, \boldsymbol{\Delta}\mathbf{u}_\mathbf{m}(t-1)]^T$, where $\boldsymbol{\Delta}\mathbf{u}_\mathbf{i}(t-1) = [\Delta u_i(t-1), \ldots, \Delta u_i(t-M)]^T$. Finally, $\mathbf{y}(t+d|t) = [y_1(t+d_1|t), \ldots, y_n(t+d_n|t)]^T$.

Equation (4.6) can also be written as $\hat{\mathbf{Y}} = \mathbf{G}\boldsymbol{\Delta}\mathbf{u} + \mathbf{f}$, where $\mathbf{f}$ represents the free response of the systems after the future instant $t+d$, i.e., the response of the systems if no changes in the control actions were

[1]The parameter M could be defined individually for each input-output response but, for simplicity, a fixed value is used for all responses.

made, and the term $\mathbf{G\Delta U}$ is the forced response, which shows how the system would react due to the future control increments. Note that the output of the predictor $\mathbf{y}(t+d|t)$ is the free response at the time instant $t+d$.

The future control action increments vector $\mathbf{\Delta U}$ is obtained by minimizing the following cost function

$$\mathbf{J} = (\hat{\mathbf{Y}} - \mathbf{Y_r})^T \mathbf{Q_y} (\hat{\mathbf{Y}} - \mathbf{Y_r}) + \mathbf{\Delta U}^T \mathbf{Q_u} \mathbf{\Delta U},$$

where $\mathbf{Q_y}$ and $\mathbf{Q_u}$ are matrices that represent the weights of the future errors and the control increments, respectively. $\mathbf{Y_r}$ is the vector of future setpoints and, considering the case where all future references are constant, it can be defined as $\mathbf{Y_r} = \mathbf{1 y_r}$, where $\mathbf{1}$ is a block diagonal matrix composed of n column vectors with dimension N_{yi}, with all entries equal to one and $\mathbf{y_r} = [y_{r1}(t), \ldots, y_{rn}(t)]^T$, and y_{ri} is the setpoint for the ith output. Without constraints[2], an algebraic solution can be found, which is given by

$$\mathbf{\Delta U} \quad = \quad \mathbf{K}(\mathbf{Y_r} - \mathbf{f}) \tag{4.7}$$

where $\mathbf{K} = (\mathbf{G}^T \mathbf{Q_y} \mathbf{G} + \mathbf{Q_u})^{-1} \mathbf{G}^T \mathbf{Q_y}$ is the control gain and $\mathbf{f} = \mathbf{H\Delta u}(t-1) + \mathbf{1}\hat{\mathbf{y}}(t+d|t))$. Since only the control increments of the current time instant t of $\mathbf{\Delta u}$ will be used, Eq. (4.7) is simplified to

$$\mathbf{\Delta u}(t) = \mathbf{K_1}(\mathbf{Y_r} - \mathbf{f}), \tag{4.8}$$

where $\mathbf{K_1}$ is an $m \times \sum_{i=1}^{n} N_{yi}$ matrix and $\mathbf{\Delta u}(t) = [\Delta u_1(t), \ldots, \Delta u_m(t)]^T$. Hence, considering Eq. (4.7), Eq. (4.8) can be rewritten as

$$\mathbf{\Delta u}(t) + \mathbf{K_1 H \Delta u}(t-1) = \mathbf{K_1 1}(\mathbf{y_r} - \mathbf{y}(t+d|t)). \tag{4.9}$$

In the frequency domain, the control actions are computed using $\mathbf{U}(z) = \mathbf{C}(z)(\mathbf{R}(z) - \mathbf{Y_d}(z))$, which is obtained after applying the $\mathcal{Z}$-Transform to Eq. (4.9). $\mathbf{Y_d}(z)$ is the transfer function representation of the output of the predictor structure, $\mathbf{R}(z)$ represents the setpoints

[2]This solution also holds for constrained problems if the constraints are not active.

and $\mathbf{C}(z)$ is the primary controller, which is given by

$$\mathbf{C}(z) = \begin{bmatrix} C_{11}(z) & C_{21}(z) & \cdots & C_{n1}(z) \\ C_{12}(z) & C_{22}(z) & \cdots & C_{n2}(z) \\ \vdots & \vdots & \ddots & \vdots \\ C_{1m}(z) & C_{2m}(z) & \ldots & C_{nm}(z) \end{bmatrix},$$

where $C_{ij}(z)$ is a transfer function representing the controller from the jth input to the ith output.

4.1.2 Obtaining the Predictor Structure

The role of the predictor structure is to obtain the expected value of the outputs after the dead time, which is given by Eq. (4.4). By inspection, it is evident that this equation is already in the format required by the predictor structure of the FSP. The first term of Eq. (4.4) is $\overline{y}_i(t + d_i|t)$, i.e., the open-loop prediction based on the step response of the system, which is delayed by d_i and appears in the second term of Eq. (4.4) as $\overline{y}_i(t|t)$, which is used to compute the prediction error. However, Eq. (4.4) can not be used directly due to the presence of the summations with infinite terms. To overcome this problem, consider the relation between the impulse response of a system and its step response:

$$y(t) = \sum_{i=1}^{\infty} h_i u(t-i) = \sum_{i=1}^{\infty} g_i \Delta u(t-i), \tag{4.10}$$

where $h_i = g_i - g_{i-1}$. As shown before, the difference $g_i - g_{i-1} \cong 0$ for a sufficiently large i. Thus, the summation in Eq. (4.4) can be truncated after M terms. Applying this equivalency to Eq. (4.4), it follows that

$$\begin{aligned} \hat{y}_i(t + d_i|t) \quad &= \quad \sum_{j=1}^{m} \left(\sum_{l=1}^{M} h_{ij,d_i+l} u_j(t-l) \right) \\ &\quad + y_i(t) - \overline{y}_i(t|t). \end{aligned} \tag{4.11}$$

Applying the $\mathcal{Z}$-Transform on every prediction of the ith output according to Eq. (4.11), and rearranging the result in matrix form results in

$$\mathbf{Y_d}(z) = (\mathbf{I} - \mathbf{L_n}(z))\mathbf{G_{DMC}}(z)\mathbf{U}(z) + \mathbf{y}(z) \tag{4.12}$$

where $\mathbf{G_{DMC}}(z)$ is an $n \times m$ matrix transfer function representing the nominal fast model based on the step response of the system used internally by the DMC algorithm, whose elements are given by

$$G_{DMC,ij}(z) = \sum_{l=1}^{M} h_{ij,d_i+l} z^{-l}. \tag{4.13}$$

If the number of coefficients M is large enough $\mathbf{G_{DMC}}(z) \cong \mathbf{G_n}(z)$. By inspection, it is easy to see that Eq. (4.12) corresponds to the MIMO predictor structure of Figure 17 with $\mathbf{F_r}(z) = \mathbf{I}$, where $\mathbf{I}$ is an n dimension identity matrix.

The transfer function matrices that describe the behaviour of the closed-loop system for changes in the set-point $\mathbf{R}(z)$, and disturbances $\mathbf{Q}(z)$ and $\mathbf{N}(z)$ are[3]

$$\mathbf{H_R}(z) = \mathbf{P_n C}(\mathbf{I} + \mathbf{G_n C})^{-1} \tag{4.14}$$

$$\mathbf{H_N}(z) = \{\mathbf{I} - \mathbf{P_n C}(\mathbf{I} + \mathbf{G_n C})^{-1}\mathbf{F_r}\}\mathbf{P_p} \tag{4.15}$$

$$\mathbf{H_Q}(z) = \{\mathbf{I} - \mathbf{P_n C}(\mathbf{I} + \mathbf{G_n C})^{-1}\mathbf{F_r}\}\mathbf{P_n} \tag{4.16}$$

Since $\mathbf{F_r}(z) = \mathbf{I}$, the original DMC algorithm is equivalent to a MIMO Smith Predictor (MIMO-SP), and as such, suffers from the same problems: (i) since $\mathbf{P_n}(z)$ and $\mathbf{P_p}(z)$ are factors in equations Eq. (4.16) and Eq. (4.15), respectively, the disturbance rejection will always be, at least, as slow as the open-loop response, and (ii) if the controller is tuned to improve the set-point tracking response, the robustness of the system will be compromised.

As was explained in Chapter 2, the addition of the prediction error filter $\mathbf{F_r}(z)$ in the MIMO-SP structure allowed an extra degree of freedom, which could be used to improve robustness or the disturbance rejection capabilities of the closed-loop system without changing the nominal reference-tracking response [1, 3]. Using the same idea, a modification in the standard DMC is proposed to make its prediction structure equivalent to an FSP, while maintaining the advantages of an MPC strategy.

[3]The dependency on the complex variable z was omitted for the sake of visualization

4.2 FILTERED DMC

In the proposed Filtered Dynamic Matrix Control (FDMC), the disturbance $\eta_i(t|t)$ is now computed by the sum of the n filtered prediction errors, i.e.,

$$\eta_i(t|t) = \sum_{j=1}^{n} F_{r,ij}(q)(y_j(t) - \overline{y}_j(t|t)) \tag{4.17}$$

where q^{-1} is the back-shift operator, i.e., $q^{-1}y(t) = y(t-1)$, and $F_{r,ij}(q)$ is a polynomial that represents the filter of the jth output prediction error considering the ith output.

For the time domain analysis, the filters $F_{r,ij}$ are represented by their impulse response, i.e.,

$$F_{r,ij}(q) = \sum_{l=0}^{\infty} h_{ij,l}q^{-l} \cong \sum_{l=0}^{M_f} h_{ij,l}q^{-l}$$

where $h_{ij,l}$ is the lth impulse response coefficient of the respective filter. Since the filters are stable, the infinite summation can be truncated after M_f terms. For implementation and design, the filters are represented by transfer functions, which are easier to work with.

Using Eq. (4.17) in Eq. (4.1), the new predictions after the minimum dead time $y_i(t + d_i + k|t)$, $\forall k > 0$, and the predictions at instant $t + d_i$ can be obtained for each output:

$$\begin{aligned} y_i(t + d_i + k|t) &= \sum_{j=1}^{m}\left(\sum_{l=1}^{\infty} g_{ij,d_i+l}\Delta u_j(t + k - l)\right) \\ &\quad + \sum_{j=1}^{n} F_{r,ij}(q)(y_j(t) - \overline{y}_j(t|t)) \end{aligned} \tag{4.18}$$

and

$$\begin{aligned} y_i(t + d_i|t) &= \sum_{j=1}^{m}\left(\sum_{l=1}^{\infty} g_{ij,d_i+l}\Delta u_j(t - l)\right) \\ &\quad + \sum_{j=1}^{n} F_{r,ij}(q)(y_j(t) - \overline{y}_j(t|t)). \end{aligned} \tag{4.19}$$

By isolating the term $\sum_{j=1}^{n} F_{r,ij}(q)y_j(t)$ of Eq. (4.19) and substituting in Eq. (4.18), the predictions after $t + d_i$ become

$$\begin{aligned} y_i(t + d_i + k|t) &= \sum_{j=1}^{m} \left(\sum_{l=1}^{k} g_{ij,d_i+l} \Delta u_j(t + k - l) \right. \\ &\quad \left. - \sum_{l=1}^{\infty} a_{ij,l} \Delta u_j(t - l) \right) \\ &\quad + y_i(t + d_i|t), \end{aligned} \tag{4.20}$$

where $a_{ij,l} = g_{ij,d_i+k+l} - g_{ij,d_i+l}$. This equation is the same as Eq. (4.5), i.e., the addition of the filtered prediction error does not alter the primary controller.

Hence, the prediction structure is given by Eq. (4.19), which, by inspection, is already in the format required by the predictor structure of the FSP. The first term of Eq. (4.20) is $\overline{y}_i(t+d_i|t)$, i.e., the open-loop prediction based on the step response of the system, which is delayed by d_i and appears in the second term of Eq. (4.20) as $\overline{y}_i(t|t)$, which is used to compute the prediction error. The error is then filtered and added to $\overline{y}_i(t + d_i|t)$ to obtain the the corrected prediction $y_i(t + d_i|t)$. Applying the $\mathcal{Z}$-Transform to Eq. (4.20) results in

$$\mathbf{Y_d}(z) = (\mathbf{I} - \mathbf{L_n}(z)\mathbf{F_r}(z))\mathbf{G_{DMC}}(z)\mathbf{U}(z) + \mathbf{y}(z), \tag{4.21}$$

which is exactly Eq. (4.12), if $\mathbf{F_r}(z) = \mathbf{I}$. The filter is given by

$$\mathbf{F_r}(z) = \begin{bmatrix} F_{r,11}(z) & \cdots & F_{r,1n}(z) \\ \vdots & \ddots & \vdots \\ F_{r,n1}(z) & \cdots & F_{r,nn}(z) \end{bmatrix}.$$

As discussed in Chapter 2, the predictor filter $\mathbf{F_r}(z)$ plays two important roles: (i) adjust the disturbance rejection response and noise attenuation, and (ii) increase the robustness [1]. It is worth noting that, although $\mathbf{F_r}(z)$ can be a full matrix transfer function, in practice a diagonal $\mathbf{F_r}(z)$ is usually enough to achieve a good compromise between performance and robustness. Also, the condition $\mathbf{F_r}(1) = \mathbf{I}$ must be true, otherwise the reference-tracking properties of the closed-loop system will be compromised.

4.2.1 Improving Disturbance Rejection

As discussed in Chapter 2, and from input-output relationships given by Eqs. (4.14) to (4.16), it can be seen that in the nominal case the filter only affects the noise and disturbance rejection responses. Thus, it can be designed to obtain a desired disturbance rejection without affecting the set-point tracking performance. In the case that $\mathbf{F_r}(z)$ is not properly designed (for example $\mathbf{F_r}(z) = \mathbf{I}$), the disturbance rejection response is governed by the open-loop dynamics of the plant $\mathbf{P_n}(z)$ and $\mathbf{P_p}(z)$ [1]. In the case of input disturbances $\mathbf{Q}(z)$, it was shown that the filter $\mathbf{F_r}(z)$ can be tuned to cancel the undesired poles of the nominal fast model $\mathbf{G_n}(z)$ inside the predictor. By doing this, these undesired poles do not affect the input disturbance rejection of the system. Unfortunately, this procedure can not be used directly with the DMC because, as was shown in Eq. (4.13), the nominal fast model used by DMC is a Finite Impulse Response (FIR) model with no poles. Hence, the above procedure not applicable in this case.

Thus, to improve the disturbance rejection another procedure is required. From Eqs. (4.14) to (4.16), it is possible to see that the closed-loop reference to output response, Eq. (4.14), appears inside the other equations multiplied by the filter $\mathbf{F_r}(z)$. Hence, two tuning guidelines can be derived, a practical guideline and an analytical one.

For the practical one, consider the fact that, ideally, the controller completely decouples the dynamics of the system, i.e., the reference to output closed-loop delay-free response of the system can be approximated by a diagonal matrix transfer function $\mathbf{G_{cl}}$ given by

$$\mathbf{G_{cl}}(z) = \mathbf{G_n C}(\mathbf{I}+\mathbf{G_n C})^{-1} \cong \begin{bmatrix} G_{cl,1}(z) & 0 & \cdots & 0 \\ 0 & G_{cl,2}(z) & \cdots & 0 \\ 0 & 0 & \cdots & G_{cl,n}(z) \end{bmatrix}.$$

Simulating the nominal closed-loop system, the dominant dynamics of each element $G_{cl,i}$ can be estimated by, for example, a first order transfer function. For the input disturbance $\mathbf{Q}(z)$, from Eq. (4.16), using the given approximation, the following approximated closed-loop dynamics is obtained

$$\mathbf{H_Q}(z) \cong (\mathbf{I} - \mathbf{L_n G_{cl} F_r})\mathbf{P_n}.$$

Since all the matrices inside the parenthesis are diagonal, the resulting transfer matrix is also a diagonal matrix, whose elements are

given by

$$1 - G_{cl,i}(z)F_{r,i}(z)z^{-d_i}. \tag{4.22}$$

Each ith element of $\mathbf{I} - \mathbf{L_n}\mathbf{G_{cl}}\mathbf{F_r}$ will be multiplied by the elements of the ith line of $\mathbf{P_n}(z)$, hence, the filters $F_{r,i}(z)$ can be tuned so that Eq. (4.22) has zeros at the undesirable poles present in the ith line of $\mathbf{P_n}(z)$, thus canceling the unwanted poles. This procedure can also be applied to the output disturbance $\mathbf{N}(z)$, whose closed-loop dynamics are given by Eq. (4.15).

The analytical method consists of the same procedure as the practical one, but without the approximation, i.e., the nominal closed-loop transfer matrix is obtained, and then the same procedure described above is used. This method is more difficult to apply when the number of variables increases but it can achieve a better disturbance rejection than the practical one because it does not use approximations. However, through simulations results, the practical method has proven to be reliable.

4.3 IMPLEMENTATION

In [14, 72, 91], it was shown the recursive properties of DMC through a state-space interpretation of the algorithm, which will be presented briefly. This recursiveness does not affect the resulting controller, it only changes the software implementation of the DMC in such a way that it is faster and more efficient. The recursive form also facilitates the inclusion of the prediction filter. The filter can be included in the original implementation, i.e., obtaining matrices $\mathbf{G}$ and $\mathbf{H}$ of Eq. (4.6), but it is much more cumbersome.

For simplicity, the SISO case will be used to describe the recursiveness of DMC. The SISO open-loop predictions at the time instant $t+k$, given the information at t and $t-1$, are

$$\overline{y}(t+k|t) = \sum_{i=k+1}^{\infty} g_i \Delta u(t+k-i) \tag{4.23}$$

$$\overline{y}(t+k|t-1) = \sum_{i=k+2}^{\infty} g_i \Delta u(t+k-i) \tag{4.24}$$

The difference of the predictions at $t+k$ at instants t and $t-1$ is only the new control increment $\Delta u(t-1)$, which was not known at $t-1$. Subtracting Eq. (4.24) from Eq. (4.23), the predictions can be

rewritten recursively:

$$\overline{y}(t+k|t) = g_{k+1}\Delta u(t-1) + \overline{y}(t+k|t-1). \tag{4.25}$$

The recursive DMC operates as follows. A vector with M elements is stored in memory, $\overline{\mathbf{y}} = [\overline{y}(t|t-1), \ldots, \overline{y}(t+M-1|t-1)]^T$, whose elements are the future predictions given the known past control moves up to instant $t-1$. During initialization at instant t_0, it can be considered that the system is at steady state and that all future predictions are equal to the current output of the system $y(t_0)$.

After the initialization, at instant t, it is necessary to update the vector since the past control move $\Delta u(t-1)$ is now known:

$$\overline{\mathbf{y}} = \overline{\mathbf{y}} + \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_M \end{bmatrix} \Delta u(t-1) \tag{4.26}$$

Hence, the vector $\overline{\mathbf{y}}$ is updated. After the update, it is necessary to move the values inside vector $\overline{\mathbf{y}}$. This is important because at the next time instant, $t+1$, the future predictions from instant $t+1$ to $t+M$ based in the control moves up to instant t will be required. Thus, the first element, $\overline{y}(t|t)$, is removed from the vector and used to calculate the current prediction error, $\eta(t|t) = y(t) - \overline{y}(t|t)$. Because of the displacement, the last value of the vector, which should be $y_o(t+M|t)$, is not known. However, in the stable case, $\overline{y}(t+M|t) \cong \overline{y}(t+M-1|t)$, thus, the new vector $\overline{\mathbf{y}}$ will be given by

$$\overline{\mathbf{y}} = \begin{bmatrix} \overline{y}(t+1|t) \\ \overline{y}(t+2|t) \\ \vdots \\ \overline{y}(t+M-1|t) \\ \overline{y}(t+M-1|t) \end{bmatrix}. \tag{4.27}$$

Therefore, the free response calculation is computed as follows in the recursive DMC

$$\mathbf{f} = \begin{bmatrix} \overline{y}(t+1|t) \\ \vdots \\ \overline{y}(t+N|t) \end{bmatrix} + \mathbf{1}\,(y(t) - \overline{y}(t|t)), \tag{4.28}$$

where $\mathbf{1}$ is an N-dimensional vector whose elements are all ones.

Since the difference between the FDMC and the DMC is how the prediction error $\eta(t|t)$ is computed, it is very straightforward how to implement the FDMC in the recursive form. It is simply a matter of adding the filtered prediction error instead of the standard error, i.e., to substitute the term $\mathbf{1}_N\left(y(t) - \overline{y}(t|t)\right)$ in Eq. (4.28) by $\mathbf{1}_N F_r(q)(y(t) - \overline{y}(t|t))$.

This implementation is actually an indirect state-space calculation of the free response [72]. This implementation can be described by the following state-space system

$$\begin{aligned}
\overline{\mathbf{y}}(t+1) &= \mathbf{M}\overline{\mathbf{y}}(t) + \mathbf{B}\Delta u(t) \\
\overline{y}(t|t) &= \mathbf{B}'\overline{\mathbf{y}}(t) \\
\mathbf{f}(t) &= \mathbf{N}\overline{\mathbf{y}}(t) + \mathbf{1}\left[F_r(q)(y(t) - \overline{y}(t|t))\right]
\end{aligned}$$

where

$$\mathbf{M} = \left[\begin{array}{c|c} \mathbf{0}_{M-1\times 1} & \mathbf{I}_{M-1} \\ \hline 0 & \mathbf{0}_{1\times M-1} \end{array}\right], \quad \mathbf{B} = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_M \end{bmatrix},$$

$$\mathbf{B}' = \begin{bmatrix} 1 & \mathbf{0}_{1\times M-1} \end{bmatrix},$$

$$\mathbf{N} = \begin{bmatrix} \mathbf{0}_{N\times 1} & \mathbf{I}_N & \mathbf{0}_{M-N-1} \end{bmatrix}.$$

Since the elements of the matrices are mainly zeros and ones, with the exception of $\mathbf{B}$, the implementation described earlier is used to avoid unnecessary matrix operations.

4.4 RESULTS

The water-methanol distillation column presented in chapter 2.4.3 will be used here to demonstrate the FDMC capabilities.

The controller is configured as follows: prediction and control horizons are set to 40 and 10 time samples for outputs and inputs, respectively. The weights are 1 for the outputs and 25 for the inputs, and the model horizon is set to $M = 120$ samples. This configuration will result in a settling time of approximately, without taking into account the dead times, 7 minutes for both outputs, as can be seen in Figure 18. Also, a white noise with zero mean and 0.01 standard deviation were added to all simulations. Notice, however, that the disturbance is con-

siderably slow, as expected since the open-loop dynamics are present in the response.

To demonstrate the capabilities of the proposed FDMC, a filter was designed to improve the disturbance rejection of the system using the practical methodology explained earlier. From the nominal output to set-point response, a diagonal first-order matrix transfer function was obtained to represent delay-free reference to output closed-loop dynamics

$$\mathbf{G_{cl}}(z) = \begin{bmatrix} \frac{0.3486}{z-0.6514} & 0 \\ 0 & \frac{0.3486}{z-0.6514} \end{bmatrix}$$

Then the filter was tuned to cancel the undesired poles, which are the poles of $\mathbf{P_p}(z)$, resulting in the following filter:

$$\mathbf{F_{r1}}(z) = \begin{bmatrix} \frac{1.5786(1-0.8440z^{-1})}{(1-0.5037z^{-1})^2} & 0 \\ 0 & \frac{1.89(1-0.8697z^{-1})}{(1-0.5037z^{-1})^2} \end{bmatrix}.$$

The simulation results can be seen in Figure 18, where the improvement in disturbance rejection time and the fact that set-point response does not change are clear. In this particular case, the disturbance rejection time was reduced from 45 to approximately 12 minutes, for both outputs.

Now, to exemplify the importance of taking into account the trade-off between the desired disturbance rejection dynamics and the robustness to modeling errors, consider that the dead times of the nominal model were incorrectly estimated, and that the real dead times are: $d_{11} = 3$, $d_{12} = 1$, $d_{21} = 5$ and $d_{22} = 5$.

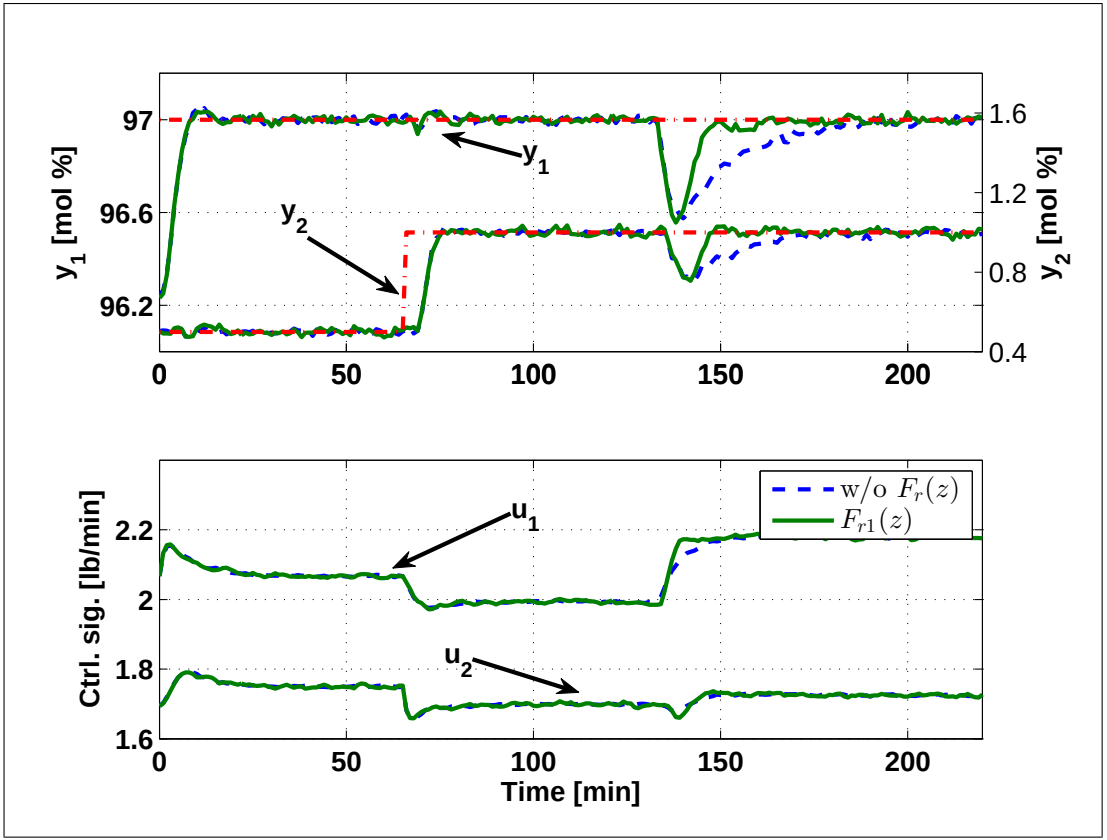

Figure 18 – Nominal response of the water-methanol process with the FDMC.

To make the robustness analysis, it is necessary to compute the uncertainty matrix $\mathbf{\Delta P}(z) = \mathbf{W_2}(z)\mathbf{\Delta}(z)\mathbf{W_1}(z)$, which is defined as[4]

$$\mathbf{\Delta P}(z) = \mathbf{P}(z) - \mathbf{P_n}(z),$$

where $\mathbf{P_n}(z)$ is the nominal model given by Eq. (2.18)[5], and $\mathbf{P}(z)$ is the actual process which differs from the nominal model only on the dead times. Considering $\mathbf{W_1}(z) = \mathbf{\Delta}(z) = \mathbf{I}$, computing this equation results in

$$\mathbf{\Delta P}(z) = \mathbf{W_2}(z) = \begin{bmatrix} \frac{-0.7440(z-1)(z+1)}{z^3(z-0.9419)} & \frac{-0.8789(z-1)(z+1)}{z^3(z-0.9535)} \\ \frac{0.5786(z-1)(z+1)}{z^7(z-0.9123)} & \frac{1.3015(z-1)(z+1)}{z^5(z-0.9329)} \end{bmatrix}. \quad (4.29)$$

The matrix $\mathbf{W_2}$ is then used to check if the robustness condition given by Eq. (2.17) is satisfied, which will indicate if the closed-loop system is stable for this particular uncertainty. From the plots shown in Figure 19, it is clear that the robustness condition is not satisfied by the filter $\mathbf{F_{r1}}(z)$ and, even if no filter is used ($\mathbf{F_r}(z) = \mathbf{I}$), the closed-loop stability is not guaranteed. Thus, for these set of uncertainties the following filter was designed (see Figure 19) to guarantee stability:

$$\mathbf{F_{r2}}(z) = \frac{0.2419(z - 0.6065)}{z - 0.9048}\mathbf{I}.$$

The response of the system considering uncertainties with and without the filter $\mathbf{F_{r2}}(z)$ are shown in Figure 20 (the step disturbance was removed from the simulation for visualization purposes), where it is clear that the filter attenuates the effects of the system uncertainties, stabilizing the process.

[4] See Section 2.4.2.

[5] The actual nominal model used by the DMC is a step-response model (see Section 4.1.2), and it is this model that should be used for the computation of the uncertainties matrix. However, for a sufficiently large model horizon M, the step-response model is almost identical to the one given by Eq. (2.18), hence, for the sake of simplicity, this one was used instead of the step-response model.

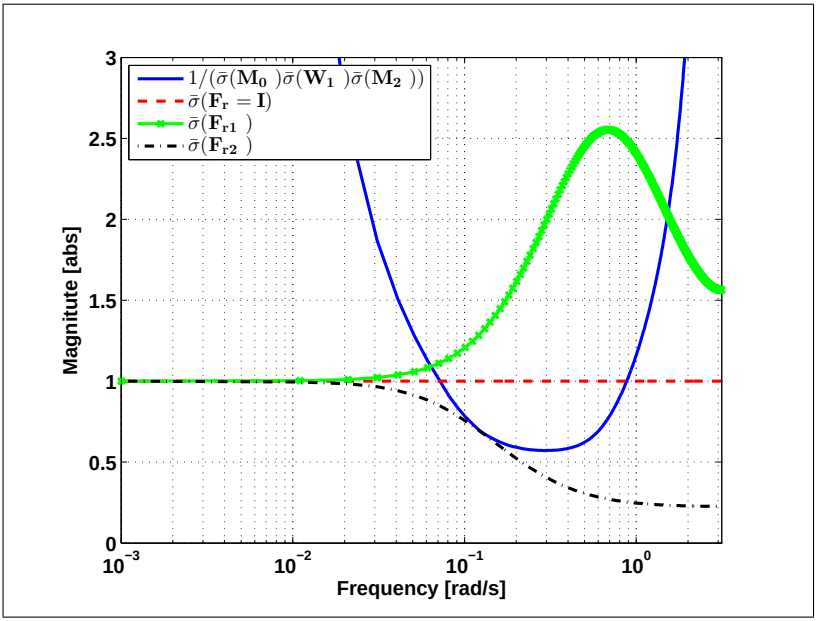

Figure 19 – Robustness condition for the MIMO example.

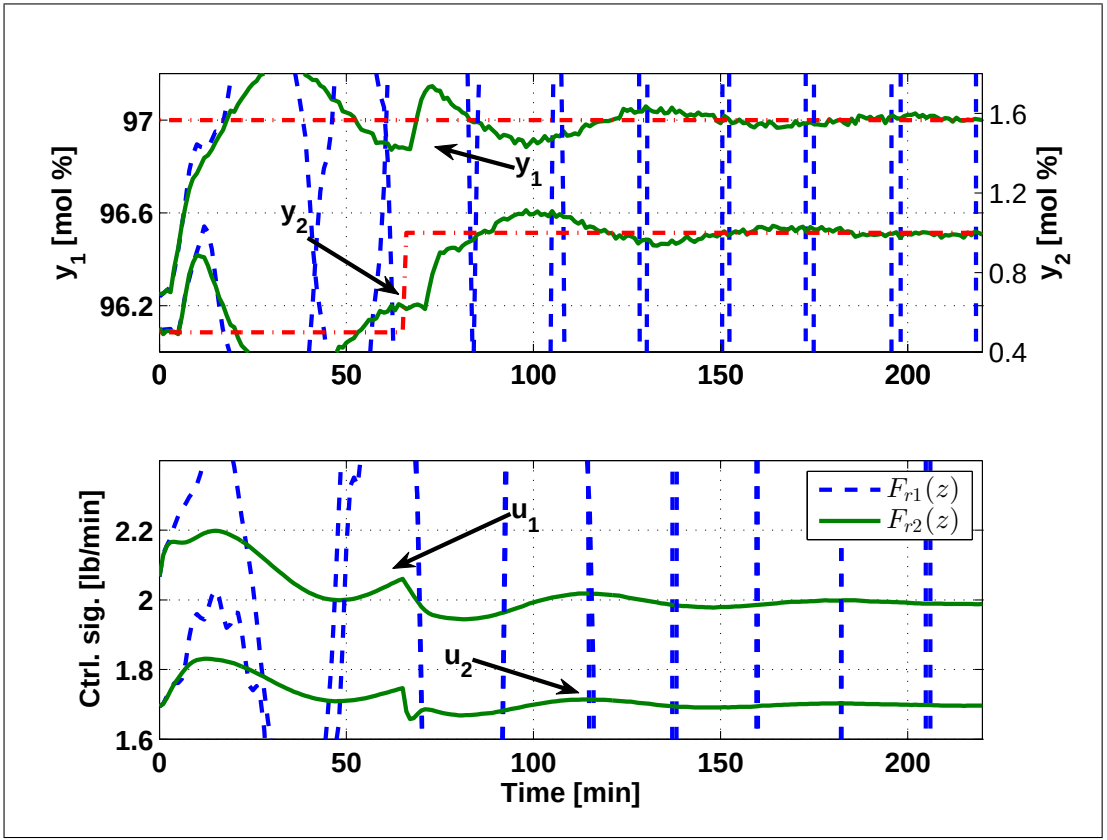

Figure 20 – Response of the water-methanol process with the FDMC and uncertainties.

4.5 FINAL COMMENTS

This chapter presented an improved industrial MPC controller based on the ideas of the Dynamic Matrix Control and of the Filtered Smith Predictor which can easily be implemented in DMC based control loops in order to improve closed-loop behavior of MIMO process with multiple time-delays. The proposed controller modifies the DMC prediction scheme by filtering the prediction errors, which alters the robustness and disturbance rejection properties of the controller without changing the reference tracking response. It is also noteworthy that the tuning methodologies of the filter are relatively simple.

5 FILTERED SMITH PREDICTOR FOR SYSTEMS WITH INPUT NONLINEARITIES

Following the ideas presented in Chapter 2 for the linear case [3], in this chapter the Nonlinear Filtered Smith Predictor (NLFSP), a dead-time compensation scheme, will be proposed to improve the robustness of dead-time systems with input nonlinearities.

The proposed NLFSP presents the same properties as the linear FSP, i.e., the nominal set-point performance is not affected by the filter, and robustness and disturbance rejection can be improved by a suitable tuning of the predictor filter [1, 34]. Note that, as in the linear case, the idea is to tune a primary controller which stabilizes the dead-time-free model and to apply this controller to the system composed by the dead-time process and the predictor.

However, at first the optimal predictor, another dead-time compensation scheme, will be discussed to establish a comparison point for the proposed algorithm.

Keep in mind that the NLFSP can be used with any controller, including linear ones. However, given the growing importance of NMPC, the simulation results with NLFSP will be done mainly with predictive controllers. This will also help to show that the implicit predictor of some NMPC can have an undesirable effect on closed-loop robustness, and that the substitution of this implicit predictor by the NLFSP can be advantageous.

The rest of the chapter is organized as follows: the general dead-time system equation with input nonlinearities will be described in Section 5.1. An useful dead-time-free auxiliary system used to study the predictors schemes will be discussed in Section 5.2. The optimal preditor and the NLFSP will be introduced and analyzed in Sections 5.3 and 5.4, respectively, and, finally, simulation results will be presented in Sections 5.5 and 5.6 to illustrate the advantages of the NLFSP.

5.1 MODELS WITH INPUT NONLINEARITIES

Nonlinear models are usually used when it is required a better representation of the system dynamics for optimal performance, e.g., it is not possible to represent variable gain or asymmetrical dynamics (changes in the gain signal) with linear models [92, 93]. In this work, the focus will be on models whose nonlinearities lie only on the system

inputs. Despite this limitation, a variety of nonlinear dynamics and static nonlinearities can be represented by this kind of model. Particularly, the commonly known Volterra and Hammerstein models are used to represent many real nonlinear processes.

The representation of nonlinear processes by Volterra series has various successful applications in process control [94–96], specially because these models allow the description of asymmetrical dynamics and gain variations of the process [93]. This model can be viewed as a generalization of the impulsive response for linear systems and its equation is given below

$$\begin{aligned} y(t) = &\sum_{i_1=1}^{\infty} h_{1i_1} u(t-i_1) \\ &+ \sum_{i_1=1}^{\infty}\sum_{i_2=1}^{\infty} h_{2i_1i_2} u(t-i_1)u(t-i_2) + \dots \\ &+ \sum_{i_1=1}^{\infty} \dots \sum_{i_m=1}^{\infty} h_{mi_1\dots i_m} u(t-i_1)\dots u(t-i_m), \end{aligned}$$

where parameters h are the coefficients of the model and m is the model order. Since this model only uses the inputs to explain the output of the process, the number of coefficients is usually very high. An alternative to this representation is to also use the information of past inputs, resulting in the Auto-Regressive Volterra (AR-Volterra) model [94], which is described by

$$\begin{aligned} A(q)y(t) = &B(q)u(t-1) \\ &+ \sum_{i_1=1}^{\infty}\sum_{i_2=1}^{\infty} h_{2i_1i_2} u(t-i_1)u(t-i_2) + \dots \\ &+ \sum_{i_1=1}^{\infty} \dots \sum_{i_m=1}^{\infty} h_{mi_1\dots i_m} u(t-i)\dots u(t-i_m), \end{aligned} \tag{5.1}$$

where q is the delay operator, i.e., $y(t)q^{-j} = y(t-j)$, $A(q) = 1+a_1q^{-1}+\dots+a_{n_a}q^{-n_a}$, and $B(q) = b_0+b_1q^{-1}+\dots+b_{n_b}q^{-n_b}$ are polynomials on q of order n_a and n_b, respectively. In practice, the infinite summations are also truncated. More information about identification and control of Volterra models can be found in [94].

The Hammerstein model is another interesting nonlinear model with input nonlinearities. It has been shown that such model structure

may account for nonlinear effects encountered in most chemical processes. It has been shown that the nonlinear behaviour of many distillation columns, pH neutralization processes, heat exchangers, as well as furnaces and reactors can be effectively modelled by a nonlinear static element followed by a linear dynamic element [97, 98]. In essence, the Hammerstein models generalize the well-known gain-scheduling concept for nonlinear control. They can be described as

$$A(q)y(t) = B(q)g(u(t-1)) \tag{5.2}$$

where polynomials $A(q)$ and $B(q)$ are the same defined for the Volterra model and $g(.) : \mathbb{R} \to \mathbb{R}$ is a function that models the static characteristics of the process gain. If $g(.)$ is chosen as a polynomial function, the Hammerstein model is actually a simplified version of the AR-Volterra model.

5.1.1 State-space representation of systems with input nonlinearities

This chapter considers the control of processes that can be modeled by a MIMO uncertain time-invariant discrete dead-time system with input nonlinearities given by

$$\begin{bmatrix} \mathbf{x_1}(t+1) \\ \vdots \\ \mathbf{x_p}(t+1) \end{bmatrix} = \mathbf{A} \begin{bmatrix} \mathbf{x_1}(t) \\ \vdots \\ \mathbf{x_p}(t) \end{bmatrix} + \begin{bmatrix} g_1(\mathbf{u}(t-d_1)) \\ \vdots \\ g_p(\mathbf{u}(t-d_p)) \end{bmatrix} + \begin{bmatrix} \mathbf{w_1}(t) \\ \vdots \\ \mathbf{w_p}(t) \end{bmatrix} \tag{5.3}$$

or, in a simplified manner, as

$$\mathbf{x}(t+1) = \mathbf{A}\mathbf{x}(t) + g(\mathbf{u}(t-D)) + \mathbf{w}(t). \tag{5.4}$$

The state vector $\mathbf{x}$ is separated in p groups in a way that it is possible to distinguish the minimum dead times from the input vector $\mathbf{u}$ to each group of states. Hence, $\mathbf{x_i} \in \mathbb{R}^{n_i}$, $\mathbf{x} \in \mathbb{R}^n$, with $n = \sum_i n_i$,

and $\mathbf{A}$ is a block diagonal square matrix[1]

$$\mathbf{A} = \begin{bmatrix} \mathbf{A_1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \mathbf{A_p} \end{bmatrix},$$

with $\mathbf{A_i}$ being an n_i dimensional square matrix, which implies that the dimension of $\mathbf{A}$ is n. The input vector is composed of m different inputs such that $\mathbf{u}(t) = [\mathbf{u_1}(t), \mathbf{u_2}(t), \ldots, \mathbf{u_m}(t)]^T$, with $\mathbf{u_i}(t) = [u_i(t), u_i(t-1), \ldots, u_i(t-l+1)]^T$, where l defines the necessary number of past inputs[2], thus, $\mathbf{u_i} \in \mathbb{R}^l$ and $\mathbf{u} \in \mathbb{R}^{m \times l}$. Also, $g_i(.)$ is a nonlinear continuous function $g_i : \mathbb{R}^{m \times l} \to \mathbb{R}^{n_i}$. The variable D represents the minimal dead time set from each group of states, $D = \{d_1, d_2, \ldots, d_p\}$, in a way that

$$g(\mathbf{u}(t-D)) \triangleq \begin{bmatrix} g_1(\mathbf{u}(t-d_1)) \\ \vdots \\ g_p(\mathbf{u}(t-d_p)) \end{bmatrix}.$$

It is worth noting that the value of l depends on the type of model, e.g., Hammerstein models requires $l = 0$ because the nonlinearity is static, while Volterra models accept $l \geq 0$. Also, $\mathbf{w}(t) = [\mathbf{w_1}(t), \ldots, \mathbf{w_p}(t)]^T$, with $\mathbf{w_i} \in \mathbb{R}^{n_i}$, and $\mathbf{w} \in \mathbb{R}^n$, is the unmeasurable additive disturbance. It is assumed that $|g_i(\mathbf{u})| = \infty \Leftrightarrow |\mathbf{u}| = \infty$, this implies that if there are unstable modes, they are present only in matrix $\mathbf{A}$.

Notice that $\mathbf{w}(t)$ can describe any kind of mismatch between the measured state at time $t+1$ and its expected value at time t (this will be better explained in Section 6.3.3), including model mismatch and unmeasurable process disturbances [6]. Its value can only be known at the time instant $t+1$, where $\mathbf{w}(t)$ is computed as

$$\mathbf{w}(t) = \mathbf{x}(t+1) - (\mathbf{A}\mathbf{x}(t) + g(\mathbf{u}(t-D))). \tag{5.5}$$

For this kind of description, by the concept of local ISS described in Appendix C, which is equivalent to the existence of a stability margin for the system [99, 100], closed-loop stability implies that $\mathbf{w}(t)$ can be bounded by a compact set[3]. The bound on $\mathbf{w}(t)$ will play an important

[1] This is better discussed through a example in Section 5.1.2.

[2] The variable l can be defined individually by input and by group of states, but, for simplicity, a fixed l is considered.

[3] Closed-loop stability guarantees that both $|\mathbf{x}(t)| < \infty$, and $|g(\mathbf{u}(t-D))| < \infty$

role in the robustness analysis of the predictor structures presented in this work. Considering that the nominal closed-loop is stable, the bigger the bound on $\mathbf{w}(t)$, the harder it is to guarantee stability.

5.1.2 Example

As described in (5.4), to separate the process into groups of states with different input delays, matrix $\mathbf{A}$ must be block diagonal. Any MIMO input-output model where the ith output depends only on its own past values and on the past inputs can be written in state-space form with a block diagonal matrix $\mathbf{A}$, where each ith group of states comprises the delayed signals of the ith output. For example, given the following Hammerstein model of a MIMO system:

$$\begin{cases} y_1(t+1) = 1.5y_1(t) - 0.56y_1(t-1) + u_1(t-d_{11}) \\ \qquad\qquad + 0.2u_1(t-d_{11})^2 + 0.1u_2(t-d_{12}) & \text{(5.6a)} \\ y_2(t+1) = 1.5y_2(t) - 0.54y_2(t-1) + 0.1u_1(t-d_{21}) \\ \qquad\qquad + 1.5u_2(t-d_{22}) - 0.3u_2(t-d_{22})^2 & \text{(5.6b)} \end{cases}$$

where the delays are $d_{11} = 3$, $d_{12} = 5$, $d_{21} = 4$, $d_{22} = 2$, then the minimal dead-time set is $D = \{3, 2\}$. To obtain the state-space representation of this system according to Eq. (5.4), first define $\mathbf{x}(t) = [\mathbf{x_1}(t)^T, \mathbf{x_2}(t)^T]^T$, where $\mathbf{x_1} = [y_1(t),\, y_1(t-1)]^T$ and $\mathbf{x_2} = [y_2(t),\, y_2(t-1)]^T$, and $\mathbf{u}(t) = [\mathbf{u_1}(t),\, \mathbf{u_2}(t)]^T$, where

$$\mathbf{u_i}(t) = [u_i(t), u_i(t-1), u_i(t-2)]^T,$$

then the system can be described as

$$\mathbf{x}(t+1) = \mathbf{A}\mathbf{x}(t) + g(\mathbf{u}(t-D)), \tag{5.7}$$

with

$$\mathbf{A} = \begin{bmatrix} 1.5 & -0.56 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1.5 & -0.54 \\ 0 & 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} \mathbf{A_1} & 0 \\ 0 & \mathbf{A_2} \end{bmatrix},$$

for all t, so that $|\mathbf{w}(t)| < \infty$ for all t. In other words, uncertainty effect is limited for non-diverging responses in such a way that $\mathbf{w}(t)$ is bounded.

i.e., $\mathbf{A}$ is block diagonal, and there are two groups of states. Also,

$$g(\mathbf{u}(t-D)) = \begin{bmatrix} u_1(t-3) + 0.2u_1(t-3)^2 + 0.1u_2(t-5) \\ 0.1u_1(t-4) + 1.5u_2(t-4) - 0.3u_2(t-2)^2 \end{bmatrix}.$$

The block-diagonal characteristic of matrix $\mathbf{A}$ is important to obtain the values of the groups of states individually after the dead time, i.e., the system can be interpreted as independent Multiple Input Single Output (MISO) systems. If this is not the case, then $p = 1$, i.e., there is only one group of states, and all the mathematical developments made in this chapter remain true. However, note that Hammerstein and Volterra models, the most used descriptions of systems with input nonlinearities, are input-output models where, in the MIMO case, each output usually depends only on the inputs and on its own past values, hence the separation of the states in groups is possible.

5.2 NONLINEAR PREDICTORS

As explained in the introduction, the use of predictors allows the project of the primary controller as if the system did not have dead time, i.e., the feedback signal that the controller uses to compute the control action is not the actual output of the system but the predicted output after the dead time calculated by the predictor block.

Hence, the signal that the primary controller perceives is the output of the following auxiliary dead-time-free system:

$$\widetilde{\mathbf{x}}(t+1) = \mathbf{A}\widetilde{\mathbf{x}}(t) + g(\mathbf{u}(t)) + \widetilde{\mathbf{w}}(t), \tag{5.8}$$

where

$$\widetilde{\mathbf{x}}(t) \triangleq \mathbf{x}(t+D|t) = \begin{bmatrix} \mathbf{x_1}(t+d_1|t) \\ \vdots \\ \mathbf{x_p}(t+d_p|t) \end{bmatrix}, \tag{5.9}$$

in a way that $\widetilde{\mathbf{x}}(t+1) = \mathbf{x}(t+D+1|t+1)$, and where $\mathbf{x}(t+D|t)$ is the prediction of the states at time $t+D$ given the information at time t, i.e, the output of the predictor at time t, and $\widetilde{\mathbf{w}}$ is an equivalent disturbance, which is related to the original one $\mathbf{w}$. The closed loop with the original system and the predictor is shown in Figure 21, where the closed loop with the auxiliary dead-time-free system is also represented. In this figure, $\mathbf{r}$ is the desired set-point, and $\kappa(\widetilde{\mathbf{x}}(t), \mathbf{r}(t))$ is the primary controller, which can be any control law, including nonlinear

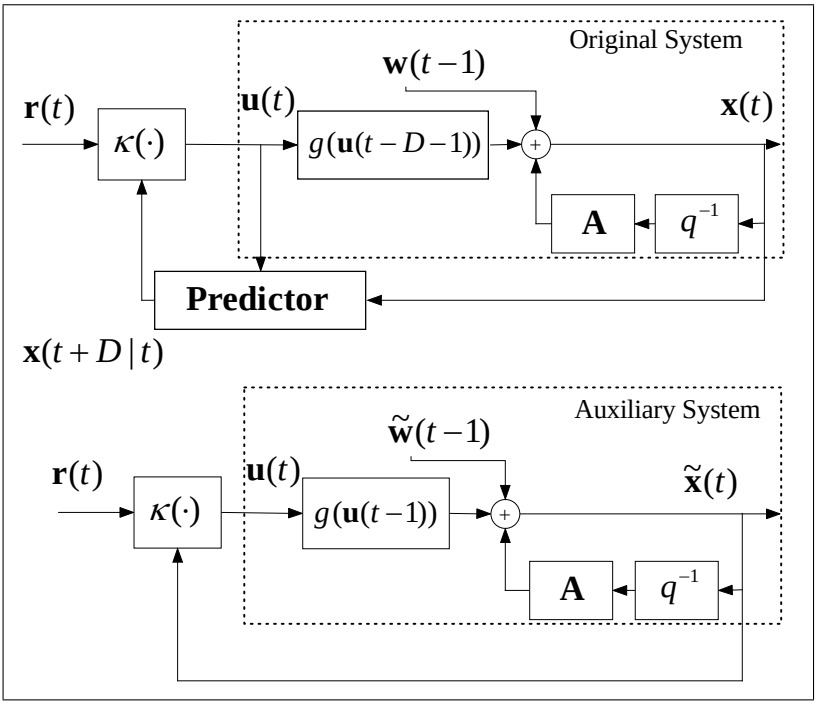

Figure 21 – Block diagrams of the closed loop system with the predictor and with the auxiliary system.

algorithms. This model is useful to analyze the effect of uncertainties on the dead-time compensator for an uncertain nonlinear system.

Since the object of study of this thesis is only the predictor structure, the following assumption is made regarding the primary controller $\kappa(.)$:

Assumption 5.1. *The primary controller* $\kappa(\tilde{\mathbf{x}}(t), \mathbf{r}(t))$ *was designed for the dead-time-free auxiliary model, which is given by Eq. (5.8), in a way that, by the local-ISS concept (see Appendix C), the closed-loop dead-time-free system remains stable if the states, inputs and disturbances are bounded, i.e.,* $\tilde{\mathbf{x}} \in \mathbb{X} = \{\tilde{\mathbf{x}} \in \mathbb{R}^n : |\tilde{\mathbf{x}}| \leq \lambda_x\}$, $\mathbf{u} \in \mathbb{U} = \{\mathbf{u} \in \mathbb{R}^{m \times l} : |\mathbf{u}| \leq \lambda_u\}$ *and* $\widetilde{\mathbf{w}} \in \widetilde{\mathbb{W}} = \{\widetilde{\mathbf{w}} \in \mathbb{R}^n : |\widetilde{\mathbf{w}}| \leq \tilde{\gamma}\}$, *where the bounds* λ_x, λ_u *and* $\tilde{\gamma}$ *are not necessarily known but are greater than zero.*

Thus, the idea of this analysis is to extend, for nonlinear processes, the predictor concepts presented for the linear SP and FSP

discussed in Chapter 2.

The disturbance $\widetilde{\mathbf{w}}$ will play an important role in the closed-loop stability of the system. The primary controller will be tuned to maintain the stability of this dead-time-free auxiliary system, hence, robustness is dependent on the bound of the disturbance $\widetilde{\mathbf{w}}$. Note, however, that the output of this auxiliary system is actually the output of the predictor. Hence, it is evident that, since each predictor uses the current information of the system ($\mathbf{x}(t)$ and $\mathbf{u}(t)$) differently to compute the future predictions, given a disturbance $\mathbf{w}$, each predictor will have a different equivalent disturbance $\widetilde{\mathbf{w}}$. Thus, the choice of the predictor structure will influence the robustness of the system. This fact will be clear in the next sections, where the optimal predictor and the proposed NLFSP will be analyzed.

5.3 OPTIMAL PREDICTORS

The optimal predictor, which is commonly found in MPC algorithms, e.g., GPC, Extended Prediction Self-Adaptive Control (EPSAC), Extended Horizon Adaptive Control (EHAC), [2] for the linear case, PNMPC [82, 83] and other MPC variations for the nonlinear case [18], will be thoroughly analyzed considering the system with input nonlinearities described in Section 5.1.

From Eq. (5.4), note that there is no effect of $\mathbf{u}(t)$ over $\mathbf{x}(t+1), \mathbf{x}(t+2), ..., \mathbf{x}(t+D)$ due to the dead time. As consequence, in absence of uncertainties, namely $\mathbf{w} = 0$, $\mathbf{x}(t+D)$ depends only on past controls, so this can be obtained knowing the current state of the plant $\mathbf{x}(t)$ and the input sequence $\mathbf{u}(t-D), \mathbf{u}(t-D+1), \ldots, \mathbf{u}(t-1)$. However, this assumption does not hold in practice because it is necessary to take into account the disturbance effects [6]. Since the future disturbance is not known, the optimal predictor estimates the disturbance using an integrated white noise model

$$\mathbf{x}(t+1) = \mathbf{A}\mathbf{x}(t) + g(\mathbf{u}(t-D)) + \frac{\mathbf{e}(t)}{\Delta}, \tag{5.10}$$

where $\mathbf{e}(t)$ is a white noise with zero mean, and $\Delta = 1 - q^{-1}$ is included to allow the representation of constant disturbances [2].

Using the auxiliary system given by Eq. (5.8), the relation between $\widetilde{\mathbf{w}}$ and $\mathbf{w}$ will be studied. It will be shown that $\widetilde{\mathbf{w}}$, using the optimal predictor, can be bigger than the original disturbance $\mathbf{w}$, which deteriorates the closed-loop robustness of the process.

5.3.1 Analysis of the optimal predictor

The optimal predictor were already analyzed in past works [8, 23], where the following theorem was used:

Theorem 5.1. *Consider a dead-time system with input nonlinearities described by Eq. (5.4), and the dead-time-free model given by Eq. (5.8). If the optimal predictor, which utilizes the model described by Eq. (5.10), is used, the equivalent disturbance is*

$$\widetilde{\mathbf{w}}(t) = \mathbf{w}(t) + \mathbf{A}'(\mathbf{w}(t) - \mathbf{w}(t-1)), \tag{5.11}$$

with

$$\mathbf{A}' = \begin{bmatrix} \sum_{j=1}^{d_1} \mathbf{A_1}^j & 0 & 0 \\ \vdots & \ddots & \vdots \\ 0 & 0 & \sum_{j=1}^{d_p} \mathbf{A_p}^j \end{bmatrix}.$$

Proof. See Appendix A.1. □

By Theorem 5.1, the equivalent disturbance of the optimal predictor is dependent on the nominal dead time of the system. The value of $\widetilde{\mathbf{w}}(t)$ will only be equal to $\mathbf{w}(t)$ if $d_i = 0, \forall i$. Hence, the robust behaviour of the system will change with the dead time.

Now, suppose that $\mathbf{w}$ is bounded in a set $\mathbb{W}$, i.e., $\mathbb{W} = \{\mathbf{w} \in \mathbb{R}^n : |\mathbf{w}| \leq \gamma\}$, then, taking the norm of Eq. (5.11),

$$\begin{aligned} |\widetilde{\mathbf{w}}(t)| &= |\mathbf{w}(t) + \mathbf{A}'(\mathbf{w}(t) - \mathbf{w}(t-1))| \\ &= |(\mathbf{I} + \mathbf{A}')\mathbf{w}(t) - \mathbf{A}'\mathbf{w}(t-1)| \\ &\leq |\mathbf{I} + \mathbf{A}'||\mathbf{w}(t)| + |\mathbf{A}'||\mathbf{w}(t-1)| \\ |\widetilde{\mathbf{w}}| &\leq (|\mathbf{I} + \mathbf{A}'| + |\mathbf{A}'|)\gamma, \end{aligned} \tag{5.12}$$

where $\mathbf{I}$ is an identity matrix of appropriate dimension.

This means that the bound on $\widetilde{\mathbf{w}}$ will always be bigger than the bound on $\mathbf{w}$, specially if the induced norm of $\mathbf{A}$ is greater than 1. Then increasing the nominal dead time also increases the norm of $\mathbf{A}'$, which results in a bigger bound for $\widetilde{\mathbf{w}}$.

With the use of dead-time compensation, it is unnecessary to take into account the dead-time length when designing the primary controller. However, for the optimal predictor, this fact is correct only from the nominal response point of view. As can be verified from Eq. (5.11), robust behaviour is closely related to the nominal dead time

length, so it is necessary to take it into account in the design procedure. It is well known that this drawback does not appear in Smith Predictor schemes, as was shown in Section 2.2.2 and in [3].

This result is consistent with the ones demonstrated for the linear case in [3], in which was shown that the robustness of the GPC algorithm, that uses the optimal predictor, is heavily dependent on the nominal dead time. This means that, for systems with considerable delay, the robustness will be compromised and noise effects will also be amplified. This problem is well known in the literature and this is usually avoided with the addition of filter polynomial $\mathbf{T}(q)$, i.e., the optimal predictor considers the following system equation

$$\mathbf{x}(t+1) = \mathbf{A}\mathbf{x}(t) + g(\mathbf{u}(t-D)) + \frac{\mathbf{T}(q)\mathbf{e}(t)}{\Delta}.$$

If correctly tuned, the addition of $\mathbf{T}(q)$ will lower the norm of $\mathbf{A}'$, thus reducing the equivalent disturbance $\widetilde{\mathbf{w}}$, though $\mathbf{A}'$ will still depend on the nominal dead time. However, as shown in [3], the effect of $\mathbf{T}(q)$ in the closed-loop robustness is not direct and its tuning procedure is not trivial. Usually, it is assumed that a stronger filtering (considered as stronger the filter with smaller bandwidth or with bigger slope if the bandwidth is the same) has better robustness properties against high-frequency uncertainties [2]. But this fact is not always true, as is shown in [101] with some counter-examples. Tuning guidelines for $\mathbf{T}(q)$ can be found in [101, 102].

5.4 NONLINEAR FILTERED SMITH PREDICTOR

In Section 5.3, it was proved that for a system with dead time, the optimal predictor amplifies the effect of unmeasured disturbances. In this section, it will be shown that if the NLFSP is used, the bound on the equivalent disturbance $\widetilde{\mathbf{w}}$ is much smaller than in the optimal predictor case, which contributes to the overall robustness of the closed-loop system. Some of the results in this section were published in [103].

The NLFSP in state-space form is shown in Figure 22, where the nominal dead-time-free system model is used to predict the output at time $t + D$, and $\mathbf{L_n}(q)$ in the figure is the minimal dead time matrix defined as

$$\mathbf{L_n}(q) = \begin{bmatrix} q^{-d_1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & q^{-d_p} \end{bmatrix}.$$

This prediction is corrected to consider the effect of unmeasurable disturbances, resulting in the following equations

$$\begin{cases} \mathbf{x_n}(t+D) = \mathbf{A}\mathbf{x_n}(t+D-1) + g(\mathbf{u}(t-1)) & \text{(5.13a)} \\ \mathbf{x}(t+D|t) = \mathbf{x_n}(t+D) + \mathbf{F_r}(q)(\mathbf{x}(t) - \mathbf{x_n}(t)), & \text{(5.13b)} \end{cases}$$

where $\mathbf{x_n}$ represents the nominal evolution of the system states, i.e., without taking into account disturbances, $\mathbf{F_r}(q) = F_r(q)\mathbf{I}$, $\mathbf{I}$ is an identity matrix and $F_r(q)$ is a stable SISO filter in the delay operator q with unitary static gain[4], i.e., $F_r(1) = 1$.

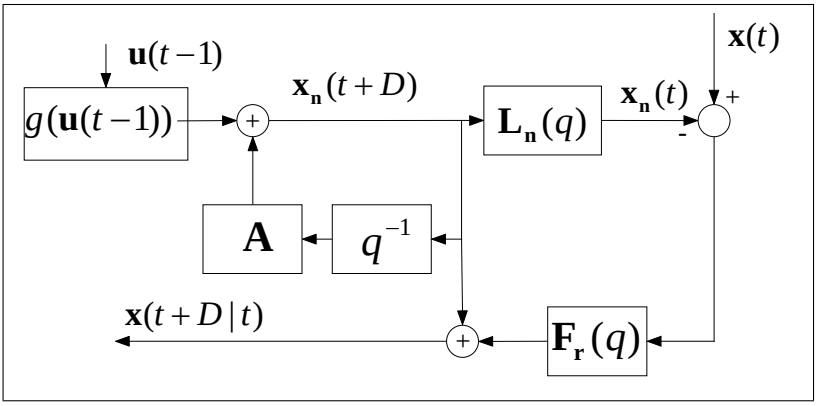

Figure 22 – Nonlinear Filtered Smith Predictor structure.

The Filtered Smith Predictor (FSP) is a general Dead-Time Compensator (DTC) structure which, in the linear case, can be used to control stable, integrating, and unstable dead-time processes [3]. The linear structure is identical to the NLFSP with the exception that function $g(.)$ is linear. The difference between the FSP and the original Smith Predictor is the addition of the filter $\mathbf{F_r}(q)$, which plays three important roles: (i) it can increase the robustness and noise attenuation, (ii) it can guarantee internal stability of the control structure,

[4]The prediction filter can also be block diagonal

$$\mathbf{F_r}(q) = \begin{bmatrix} \mathbf{F_{r1}}(q) & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \mathbf{F_{rp}}(q) \end{bmatrix},$$

with $\mathbf{F_{ri}}(q) = F_{ri}(q)\mathbf{I}$, where $\mathbf{F_{ri}}(q)$ has the dimension of the ith group of states, i.e., each group can have an independent prediction filter.

and (iii) it adjusts the disturbance rejection response [1]. The robustness characteristics of the NLFSP will be analyzed in Section 5.4.1. In Sections 5.4.2 and 5.4.3 the internal stability problem of the predictor will be discussed, and in Section 5.4.4 the disturbance rejection will be dealt with.

5.4.1 Analysis of the NLFSP

For the linear case, it was shown that if the predictor filter has low-pass characteristics, the robustness of the system is increased [1, 3]. This property also holds for the proposed NLFSP. To prove this statement, the following theorem is used.

Theorem 5.2. *Consider a dead-time system with input nonlinearities described by Eq. (5.4), and the dead-time-free model given by Eq. (5.8). If the NLFSP structure is used, the equivalent disturbance is*

$$\widetilde{\mathbf{w}}(t) = \mathbf{F_r}(q)\mathbf{w}(t), \tag{5.14}$$

with $\mathbf{F_r}(q) = F_r(q)\mathbf{I}$*, where* $F_r(q)$ *is a SISO filter.*

Proof. See Appendix A.2. □

With Eq. (5.14), the fact that the equivalent disturbance of the dead-time-free model of the system with input nonlinearities using the NLFSP is the original disturbance filtered by $\mathbf{F_r}(q)$ is proven. Notice that the filter must have unitary static gain, otherwise the system will not be able to reject constant disturbances because, at steady-state, the value of the equivalent disturbance will be $\widetilde{\mathbf{w}}(t) = \mathbf{K_f}\mathbf{w}(t)$, where $\mathbf{K_f}$ is the filter static gain, i.e., the controller will perceive a different value of the disturbance.

Given the results of Theorem 5.2, the bound on the equivalent disturbance will be analyzed. If no filter is used, i.e., $\mathbf{F_r}(q) = \mathbf{I}$, $\widetilde{\mathbf{w}} = \mathbf{w}$ and it is trivial to see that the bound on $\widetilde{\mathbf{w}}$ is the same as the one for $\mathbf{w}$. For $\mathbf{F_r}(q) \neq \mathbf{I}$, consider that the filter can be represented by its impulsive response

$$\mathbf{F_r}(q) = \sum_{j=0}^{\infty} \mathbf{H_j} q^{-j}$$

where $\mathbf{H_j} = h_j\mathbf{I}$, and h_j is a scalar and the jth coefficient of the impulse

response of the filter $F_r(q)$. Taking the norm of Eq. (5.14)

$$|\widetilde{\mathbf{w}}(t)| = \left| \sum_{j=0}^{\infty} \mathbf{H_j} \mathbf{w}(t-j) \right| \leq \sum_{j=0}^{\infty} |\mathbf{H_j}||\mathbf{w}(t-j)|,$$

and given that $\mathbf{w}$ is bounded, i.e., $|\mathbf{w}| < \gamma$,

$$|\widetilde{\mathbf{w}}(t)| \leq \gamma \left(\sum_{j=0}^{\infty} |\mathbf{H_j}| \right). \tag{5.15}$$

By definition $F_r(1) = 1$, thus, if the impulse response of the filter $F_r(q)$ has monotonic behavior, i.e., has real poles and does not have dominant zeros, the summation in Eq. (5.15) equals $\mathbf{I}$, [5] hence,

$$|\widetilde{\mathbf{w}}| \leq \gamma. \tag{5.16}$$

Therefore, it was shown that the NLFSP with an appropriate filter tuning maintains the bound on $\mathbf{w}$, which results in a more robust behaviour, if compared to the optimal predictor. Furthermore, since $\mathbf{w}(t)$ can represent model mismatch dynamics and noise, and that these type of signals are usually in the range of medium and high frequencies, when $F_r(q)$ is tuned as a low-pass filter their effects will be attenuated with the use of the filter, improving the overall robustness.

5.4.2 Stability of the Predictor Structure

The internal stability problem arises if the nominal model is not open-loop stable, which is reflected in the presence of eigenvalues outside of, or exactly at, the unit circle in matrix $\mathbf{A}$. For linear systems, internal stability depends on the prediction, namely $\mathbf{x}(t+D|t)$. This problem and its solution are well documented in various works for the linear case [1, 3, 34, 104], and they will be extended to dead-time systems with input nonlinearities.

Firstly, note that $|g(\mathbf{u})| = \infty$ if, and only if, $|\mathbf{u}| = \infty$. Then, defining an intermediate signal $\overline{\mathbf{u}}(t) = g(\mathbf{u}(t))$,[6] using Eq. (5.9), and

[5] The summation of the impulse response coefficients of a transfer function is better discussed in Appendix B.

[6] Note that $\overline{\mathbf{u}}(t)$ is bounded if the control signal is bounded, so that it is possible to compute the $\mathcal{Z}$-Transform of $\overline{\mathbf{u}}(t)$.

substituting them in Eq. (5.13a) and Eq. (5.13b), which are the predictor equations, results in

$$\begin{aligned}\mathbf{x_n}(t+D) =& \mathbf{A}\mathbf{x_n}(t+D-1) + \overline{\mathbf{u}}(t-1)\\ \widetilde{\mathbf{x}}(t) =& \mathbf{x_n}(t+D) + \mathbf{F_r}(q^{-1})(\mathbf{x}(t) - \mathbf{x_n}(t)).\end{aligned}$$

Since these equations are linear, the $\mathcal{Z}$-Transform can be used to analyze this part of the system:

$$\mathbf{L_n}(z)^{-1}\mathbf{X_n}(z) = \mathbf{L_n}(z)^{-1}z^{-1}\mathbf{A}\mathbf{X_n}(z) + z^{-1}\overline{\mathbf{U}}(z) \tag{5.17}$$

$$\widetilde{\mathbf{X}}(z) = \mathbf{L_n}(z)^{-1}\mathbf{X_n}(z) + \mathbf{F_r}(z)(\mathbf{X}(z) - \mathbf{X_n}(z)) \tag{5.18}$$

with

$$\mathbf{L_n}(z)^{-1} = \begin{bmatrix} z^{d_1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & z^{d_p} \end{bmatrix}.$$

Rearranging Eq. (5.17),

$$\mathbf{X_n}(z) = \mathbf{L_n}(z)(\mathbf{I} - z^{-1}\mathbf{A})^{-1}z^{-1}\overline{\mathbf{U}}(z), \tag{5.19}$$

and substituting this last equation in Eq. (5.18), the following is obtained after some rearrangements

$$\widetilde{\mathbf{X}}(z) = \mathbf{S}(z)\overline{\mathbf{U}}(z) + \mathbf{F_r}(z)\mathbf{X}(z), \tag{5.20}$$

where $\mathbf{S}(z) = (\mathbf{I} - \mathbf{L_n}(z)\mathbf{F_r}(z))(\mathbf{I} - z^{-1}\mathbf{A})^{-1}z^{-1}$.

From the BIBO stability condition, given that the input signal is bounded, $|g(\mathbf{u})| < \infty$, the predictor, given by Eq. (5.20), is BIBO stable if, and only if, $\mathbf{F_r}(z)$ and $\mathbf{S}(z)$ are stable. By definition $\mathbf{F_r}(z)$ is stable, thus, only the stability of $\mathbf{S}(z)$ remains. The poles of the nominal system model are defined in the term $(\mathbf{I} - z^{-1}\mathbf{A})^{-1}$ of $\mathbf{S}(z)$, so, to cancel them, the term $(\mathbf{I} - \mathbf{L_n}(z)\mathbf{F_r}(z))$ must have zeros at the undesirable locations, which is exactly the same condition as the one presented for the linear MIMO-FSP case by Eq. (2.15) in Section 2.4. Thus, the filter $\mathbf{F_r}(z)$ must be tuned to cancel the unstable poles of the system. More information about the the tuning procedures of the filter can be found in [3, 34].

Therefore, for open-loop unstable dead-time systems, the NLFSP structure must be implemented in the form presented in Figure 23, where the linear part of the system is shown more clearly.

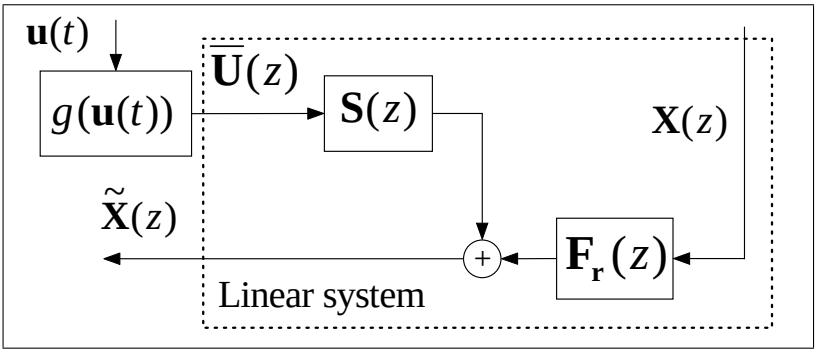

Figure 23 – Implementation of the NLFSP structure for unstable systems.

5.4.3 Equivalent Representation

Now, an equivalent description is presented which is obtained from the composition of the nonlinear system with input delay and the dead-time compensator. This representation is useful to show the role of the signal $\widetilde{\mathbf{w}}(t)$ with respect to robust stability and the offset-free prediction property of the NLFSP.

Firstly, it is necessary to define the prediction error as follows:

$$\mathbf{e}(t+D) = \mathbf{x}(t+D) - \widetilde{\mathbf{x}}(t) \tag{5.21}$$

which represents the difference between the real state, $\mathbf{x}(t+D)$, and its respective prediction obtained at time t, namely $\widetilde{\mathbf{x}}(t) = \mathbf{x}(t+D|t)$. Then, the D-steps ahead original system can be represented by:

$$\widetilde{\mathbf{x}}(t+1) = \mathbf{A}\widetilde{\mathbf{x}}(t) + g(\mathbf{u}(t)) + \widetilde{\mathbf{w}}(t) \tag{5.22}$$

$$\mathbf{x}(t+D) = \widetilde{\mathbf{x}}(t) + \mathbf{e}(t+D). \tag{5.23}$$

For presentation simplicity, consider null initial conditions for $\widetilde{\mathbf{x}}(t)$. In this case, by applying the $\mathcal{Z}$-Transform to Eq. (5.21), the following expression for $\mathcal{Z}\{\mathbf{e}(t)\}$ is obtained:

$$\mathbf{L_n}(z)^{-1}\mathbf{E}(z) = \mathbf{L_n}(z)^{-1}\mathbf{X}(z) - \widetilde{\mathbf{X}}(z)$$

$$\mathbf{E}(z) = \mathbf{X}(z) - \mathbf{L_n}(z)\widetilde{\mathbf{X}}(z)$$

Then, by using Eq. (5.18), the following expression is obtained:

$$\begin{aligned}\mathbf{E}(z) &= \mathbf{X}(z) - (\mathbf{X_n}(z) + \mathbf{L_n}(z)\mathbf{F_r}(z)(\mathbf{X}(z) - \mathbf{X_n}(z))) \\ &= (\mathbf{I} - \mathbf{L_n}(z)\mathbf{F_r}(z))(\mathbf{X}(z) - \mathbf{X_n}(z))\end{aligned} \tag{5.24}$$

Applying the $\mathcal{Z}$-transform to the system Eq. (5.4) with

$$\overline{\mathbf{u}}(t - D) = g(\mathbf{u}(t - D)),$$

results in:

$$\mathbf{X}(z) = (\mathbf{I} - z^{-1}\mathbf{A})^{-1}z^{-1}[\mathbf{L_n}(z)\overline{\mathbf{U}}(z) + \mathbf{W}(z)]$$

Substituting this equation and Eq. (5.19) in Eq. (5.24), and after some cancelations, the relationship between the prediction error and the original additive disturbance is obtained:

$$\begin{aligned}\mathbf{E}(z) &= (\mathbf{I} - \mathbf{L_n}(z)\mathbf{F_r}(z))(\mathbf{I} - z^{-1}\mathbf{A})^{-1}z^{-1}\mathbf{W}(z) \\ &= \mathbf{S}(z)\mathbf{W}(z)\end{aligned} \tag{5.25}$$

where $\mathbf{S}(z)$ was defined in Eq. (5.20).

Hence, it becomes clear that if $\mathbf{S}(z)$ is BIBO stable, then bounded disturbances result in bounded prediction errors. Moreover, in the presence of constant disturbances, if $\mathbf{F}_r(1) = \mathbf{I}$, then $\mathbf{S}(1) = \mathbf{0}$, and $\mathbf{e}(t) \to \mathbf{0}$ from final value theorem as a consequence. As previously discussed, if $\mathbf{S}(z)$ is not BIBO stable, an arbitrary small disturbance would result in an unstable response for $\mathbf{x}(t)$. However, as $\mathbf{S}(z)$ should be BIBO stable, if the system is in a stable equilibrium point, then, the relationship from any input signal to any output signal is bounded for sufficiently small inputs, guaranteeing internal stability.

As can be verified from Eq. (5.22) and Eq. (5.23), the original closed-loop representation and its equivalent description are presented in Figure 24 where $\kappa(\widetilde{\mathbf{x}}(t), \mathbf{r}(t))$ represents an arbitrary control law (linear or nonlinear).

Several points can be observed from this equivalent representation: (i) as could be expected, control law should be designed to guarantee robust stability of the system without dead time; (ii) prediction error, $\mathbf{e}(t)$, is related to output performance (disturbance rejection or set-point tracking), but has no effect over the equivalent control-loop; (iii) for the equivalent control loop, robust stability depends on the disturbance $\widetilde{\mathbf{w}}(t)$, which is defined by the robustness filter in the

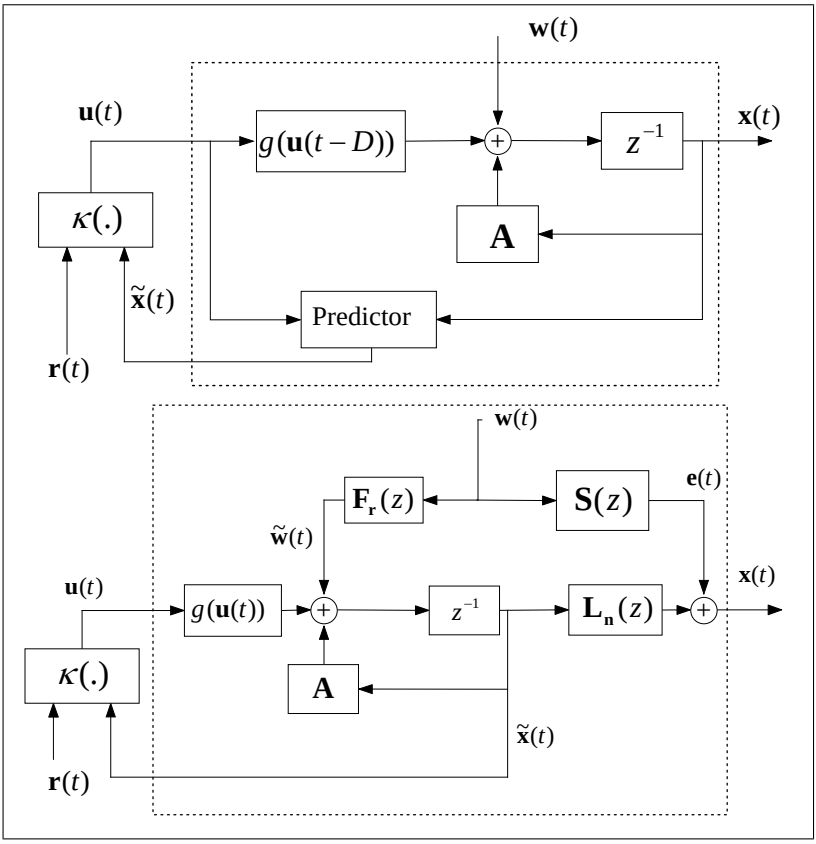

Figure 24 – Original closed-loop with dead time compensator and equivalent NLFSP description.

NLFSP approach.

In this representation, prediction error relates robust stability of the equivalent loop with the original loop. Thus, if (i) $\mathbf{S}(z)$ is a BIBO stable filter, (ii) $\mathbf{F_r}(1) = \mathbf{I}$, and (iii) if $\widetilde{\mathbf{x}}^*$ is an asymptotically stable equilibrium point for the equivalent loop subject to $\mathbf{u}(t) = \kappa(\widetilde{\mathbf{x}}(t))$ and $\widetilde{\mathbf{w}}(t) = \mathbf{F_r}(q)\mathbf{w}(t)$, then $\mathbf{x}^* = \widetilde{\mathbf{x}}^*$ is also an asymptotically stable equilibrium point for the original system. Moreover, if the eigenvalues of $\mathbf{A}$ are inside the unit circle, then $\mathbf{F_r}(z)$ does not depend on D, hence, the nominal dead time has no effect from robust stability point of view. This result is important since it directly extends the linear

FSP robustness results for systems with input nonlinearities.

5.4.4 Improving Disturbance Rejection

In the linear case, when the Smith Predictor is used, the open-loop poles of the system appear in the disturbance rejection transfer function, causing slow disturbance rejection responses when the open-loop response is slow, as was shown in [3].

This problem can be solved using the prediction filter of the FSP, which eliminates these open-loop poles from the closed-loop disturbance response through an implementation of the predictor structure which does not have these slow poles (the same is valid for unstable poles). It is not possible to use this idea directly in the nonlinear case because the concept of poles and zeros only apply to linear systems. However, from the equivalent structure shown in Figure 24, note that the effect of $\mathbf{w}(t)$ on the output $\mathbf{x}(t)$ is a combination of how the closed-loop of the dead-time-free model rejects the disturbance and also the addition of $\mathbf{w}(t)$ filtered by $\mathbf{S}(z)$ directly at the output. From Eq. (5.20), remember that the poles of $\mathbf{S}(z)$ are a combination of the poles of the filter $\mathbf{F_r}(z)$ and the eigenvalues of matrix $\mathbf{A}$. Hence, even if the closed-loop of the dead-time-free model rejects the disturbance according to specifications, the response will still be dominated by the open-loop dynamics given by $\mathbf{S}(z)$.

It is possible to avoid this problem using the same concepts used to guarantee the internal stability of the predictor explained in Section 2.4, i.e., tune the filter $\mathbf{F_r}(z)$ in such a way that Eq. (2.15) has a zero in all the undesirable open-loop locations; this will eliminate the unwanted poles of $\mathbf{S}(z)$.

Be reminded that using this procedure to improve the disturbance rejection response usually results in a filter that modifies and potentially deteriorates the closed-loop robustness, which is a natural trade-off [34].

5.4.5 Improving Robustness

In the linear case, to improve the robustness of the closed-loop system, as seen in Section 2.2.2, the first step was to obtain an estimation of the model uncertainties in the frequency domain, which usually resulted in a transfer function. Then, the prediction error filter

$\mathbf{F_r}(z)$ should be tuned such that the robustness condition given by the Nyquist stability criterion (Eq. (2.8)) would be satisfied. This generally requires a low-pass filter, which attenuates the medium and high frequency components of the uncertainties.

However, in the nonlinear case, it is not possible to use the Z-Transform to represent modelling errors in the frequency domain using a transfer function. Nonetheless, it is possible to estimate the frequency characteristics of the uncertainties using a Fourier analysis of the signal $\mathbf{w}$.

Usually, in the identification process, a family of models is obtained, and then one is chosen as the nominal one. One method to estimate the frequency characteristics of $\mathbf{w}$ is to simulate all models for a given set of inputs, and then compare the output with the expected nominal output. Through the use of the following equation, derived from Eq. (5.4), $\mathbf{w}$ can be computed

$$\mathbf{w}(t-1) = \mathbf{x}(t) - \mathbf{A}\mathbf{x}(t-1) + g(\mathbf{u}(t-D)).$$

Each comparison of the ith model with the nominal model will result in a disturbance set $\mathbf{w_i}$. Through the Fourier analysis of all these sets, the main frequency components of the uncertainties can be estimated.

By the results of Theorem 5.2, $\widetilde{\mathbf{w}} = \mathbf{F_r}(z)\mathbf{w}$, hence, if the prediction filter is designed in such a way that the medium and high frequency components of $\mathbf{w}$ are attenuated, then the robustness will increase because $\widetilde{\mathbf{w}}$ will not be affected by these components.

Another approach, which will be used in Section 5.6, is to obtain an estimation of $\mathbf{w}$ through the Fourier analysis of experimental data, i.e., to compare the nominal model output with the process data. This procedure is simpler because it needs only the nominal model but, on the other hand, it also requires experimental data in which the process were adequately excited, which can be difficult to obtain.

5.5 SIMULATION EXAMPLES

Two simple simulation examples are used here to illustrate the relationship between robustness, dead-time compensation scheme and nominal dead time length. First, a stable SISO FOPDT system with input saturation is used as this kind of simplified description is commonly used in process control problems. The second example changes the process to an unstable SISO FOPDT system with input saturation.

5.5.1 Stable case

Four simulation scenarios are considered based on the following transfer functions:

$$P_1(s) = \frac{(1+\delta K)}{(1+\delta\tau)s+1} e^{-(L_1+\delta L)s}$$
$$P_2(s) = \frac{(1+\delta K)}{(1+\delta\tau)s+1} e^{-(L_2+\delta L)s}$$

where L_i is the nominal delay, and δL, δK and $\delta\tau$ represent the dead-time, static gain, and time-constant uncertainties, respectively. In both examples, δL can assume the values of 0 [s] (nominal case) or 0.1 [s] (uncertain case). Due to the saturated input, state-space realization can be described as follows[7]:

$$\dot{x}(t) = -x(t) + \text{sat}(u(t-L_i)) + w_c(t),$$

where $L_1 = 0.2$ [s], $L_2 = 2$ [s] and $w_c(t)$ may represent input and output step disturbances and the dead-time estimation error effect. In all simulation scenarios, saturation constraint is given by $|u(t)| < 1.1$.

The discrete-time representation is given by:

$$x(t+1) = \big[a\big]\, x(t) + \big[(1-a)\big]\, \text{sat}(u(t-d_i)) + w(t), \tag{5.26}$$

where $a = e^{-T_s/\tau} = e^{-T_s/1}$, $d_i = L_i/T_s$, $T_s = 0.1$ [s] is the sampling period and $w(t) = x(t+1) - ax(t) - (1-a)\text{sat}(u(t-d_i))$.[8]

To control this process, three control structures will be used: GPC, a 2DOF PI with the NLFSP, and a 2DOF PI with the optimal predictor[9]. In all cases, the primary controller is such that Assumption 5.1 holds.

[7] With a slight abuse of notation by also using t to represent the continuous time.

[8] Note that w is not directly obtained as a sampled version of w_c, despite the fact that it describes the effect of w_c in discrete time.

[9] The FSP is not compared to the well known Dynamic Matrix Control (DMC) strategy because, as shown in Chapter 4, the DMC implicitly defines an SP dead-time compensation scheme with $\mathbf{F_{rdmc}}(z) = \mathbf{I}$, for any dead-time length, hence it also inherits the same characteristics of the original SP approach.

5.5.1.1 GPC strategy

The GPC strategy is defined by means of the minimization of the following cost function:

$$J = \sum_{j=1}^{N} [(r(t) - x(t+d+j|t))^2 + Q_u(\Delta u(t+j-1))^2]$$

with $N = 30$, $Q_u = 0.5$, and $\Delta u(t+j) = u(t+j) - u(t+j-1)$.

The input restriction is also considered in the minimization procedure.

5.5.1.2 2DOF PI with optimal predictor

The discretized Proportional-Integral (PI) controller with an anti-windup scheme is defined by the following control law:

$$u(t) = u_p(t) + u_i(t), \tag{5.27}$$

$$u_p(t) = k_p e(t), \tag{5.28}$$

$$u_i(t) = u_i(t-1) + k_i T_s [e(t) + k_a(\text{sat}(u(t-1)) - u(t-1))] \tag{5.29}$$

where $k_p = 3.4$, $k_i = 11.3$, $k_a = 1$, $e(t) = r_f(t) - y(t+d_i|t)$ is the output error, and $r_f(t)$ is the set-point $r(t)$ filtered by the reference filter $F(z)$, which is defined as

$$F(z) = \frac{k_i T_s z}{[(k_p + k_i T_s)z - k_p]}.$$

In this case, the 2DOF PI controller parameters (k_p and k_i) and reference filter were purposely obtained from the GPC unconstrained solution. This can be done because the GPC uses implicitly an FSP structure, where the parameters of the filter $F_r(z)$ depend on the nominal dead-time, and the parameters of the primary controller depend on the horizons and weights of the cost function [2]. Hence, in the absence of disturbance and active constraints, this 2DOF PI and the proposed GPC present the same close-loop behavior.

To compute the optimal prediction, the disturbance is approxi-

mated by an integrated white noise, as explained earlier,

$$x(t+1) = \big[a\big]\, x(t) + \big[(1-a)\big]\, \mathrm{sat}(u(t-d_i)) + \frac{e(t)}{\Delta}.$$

Multiplying both side of this equation by Δ, and after some rearrangements,

$$\begin{aligned} x(t+1|t) =& [a]x(t|t) + x(t|t) \\ & [(1-a)]\mathrm{sat}(u(t-d_i)) + [(1-a)]\mathrm{sat}(u(t-d_i-1)), \end{aligned}$$

which can also be described in an augmented (incremental) prediction model:

$$\begin{aligned} \boldsymbol{\xi}(t+1) =& \begin{bmatrix} a+1 & -a \\ 1 & 0 \end{bmatrix} \boldsymbol{\xi}(t) \\ & + \begin{bmatrix} (1-a) \\ 0 \end{bmatrix} \{\mathrm{sat}(u(t-d_i)) - \mathrm{sat}(u(t-1-d_i))\} \\ y(t) =& \begin{bmatrix} 1 & 0 \end{bmatrix} \boldsymbol{\xi}(t), \end{aligned}$$

where $\boldsymbol{\xi}(t) = [x(t|t), x(t-1|t)]^T$.

5.5.1.3 2DOF PI with NLFSP

The primary controller and reference filter remains the same as in the case of the PI with the optimal predictor. The only change is the predictor, which is now the NLFSP. In order to obtain a fair comparison between controllers, the filter $F_r(s)$ of the NLFSP was tuned to obtain the same nominal response as the GPC for the case $L_1 = 0.2$ [s]. This gives

$$F_r(z) = \frac{1.2967(z - 0.8083)}{z - 0.7515}.$$

This filter presents a high-pass behavior, but it is equivalent to the GPC unconstrained solution for $L_1 = 0.2$ [s][10].

5.5.1.4 Simulation Results

Output responses and control signals, for the case without dead-time estimation errors, are presented in Figures 25 and 26. Note that re-

[10]Details about the robustness filter of the GPC can be found in [34].

ference tracking responses are equivalent in all scenarios without dead-time estimation uncertainties. Also, the disturbance rejection responses are similar for GPC and PI+NLFSP when $L_1 = 0.2$ [s], as expected. The disturbance rejection with the optimal predictor is more aggressive in this case.

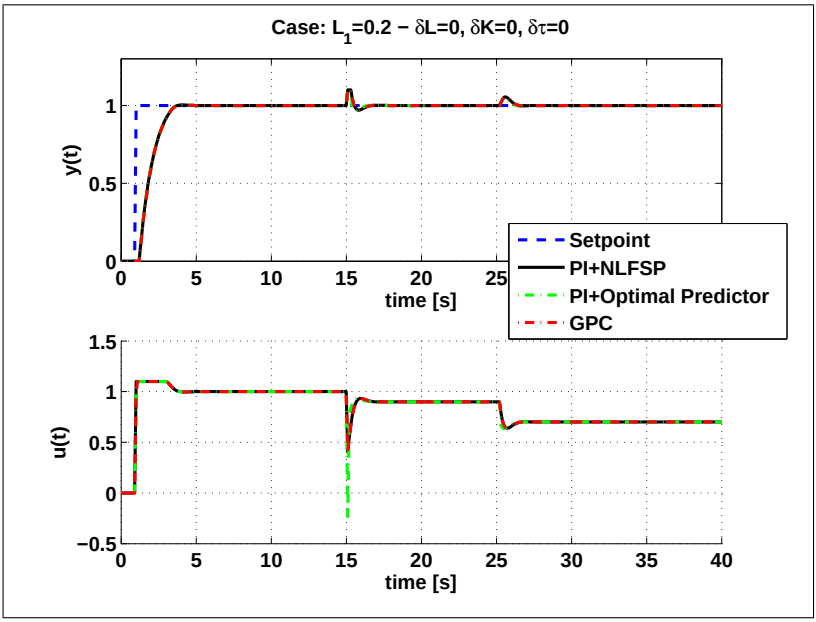

Figure 25 – Stable Case ($L = 0.2$): Output responses and control signals for the nominal system.

In the presence of uncertainties with $\delta k = 0.1$, $\delta L = 0.1$, and $\delta\tau = 0.1$ (Figures 27 and 28), only the GPC and PI+NLFSP remain stable in closed-loop when $L_1 = 0.2$ [s]. The PI with optimal predictor responses suffers from a high oscillatory behaviour which is an indicative of robustness problems. For $L_2 = 2$ [s], only the PI+NLFSP strategy remains stable with a smooth response because the robustness was not modified in this new scenario. This is an undesired characteristic for both the optimal predictor and the GPC, as robustness margins clearly depend on the nominal dead-time length.

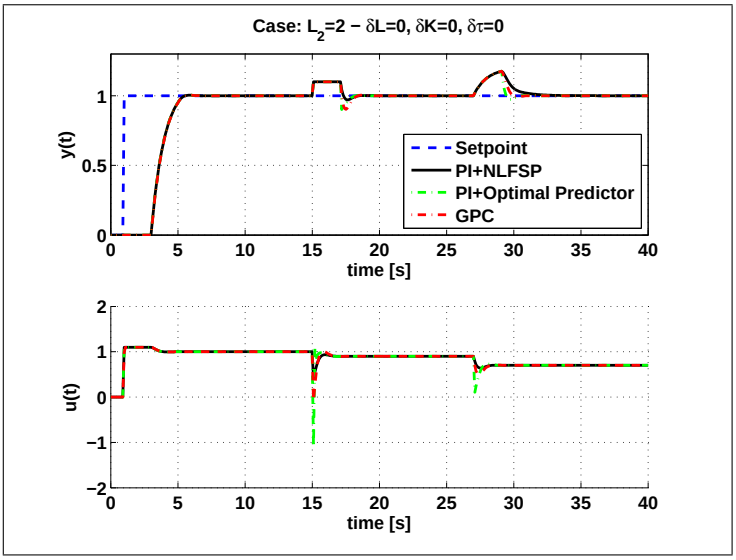

Figure 26 – Stable Case ($L = 2$): Output responses and control signals for the nominal system.

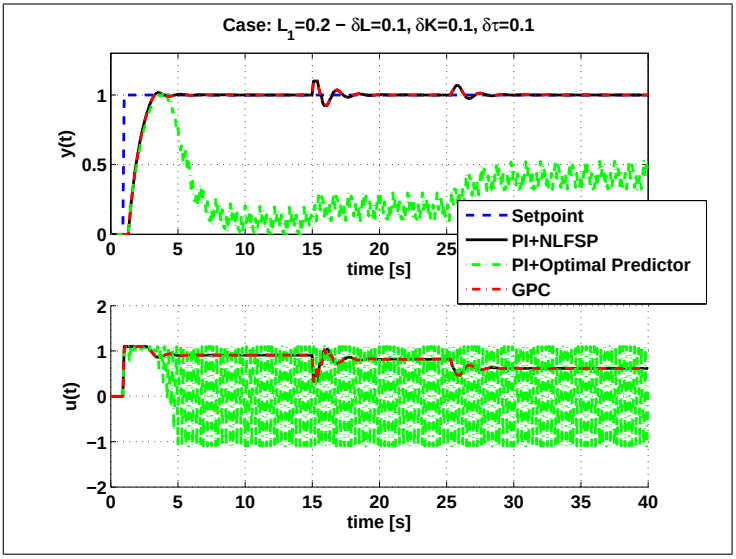

Figure 27 – Stable Case ($L = 0.2$): Output responses and control signals for the uncertain system.

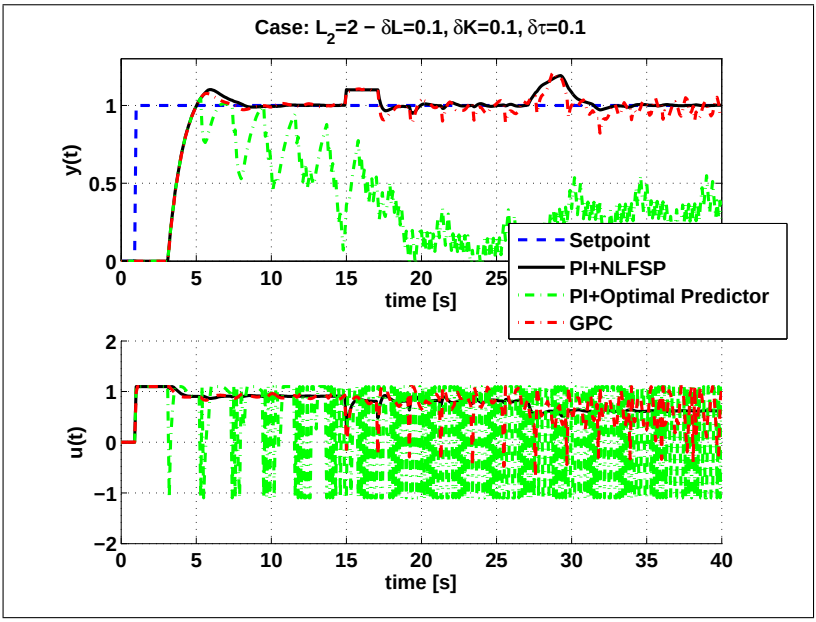

Figure 28 – Stable Case ($L = 2$): Output responses and control signals for the uncertain system.

With respect to the NLFSP, robust stability is not related to the nominal dead-time length. The price to pay is that, if the same robustness filter is used, disturbance rejection responses become slower for longer delays when compared with the optimal predictor, which indicates a natural trade-off. Note that $\delta L = 0.1$ [s], represents 50% of the nominal delay when $L_1 = 0.2$ [s] and only 5% when $L_2 = 2$ [s]. This fact is important to illustrate the advantage of the NLFSP since the main controller can be defined neglecting nominal dead-time for both, nominal and robustness behaviours. This fact is not true for the GPC, for instance, as robust behaviour is sensitive to nominal dead-time.

5.5.2 Unstable case

Following the idea of the stable case, a simulation example based on a simple FOPDT unstable description is presented. In this case, it will be shown that FSP can be used to improve robustness of systems

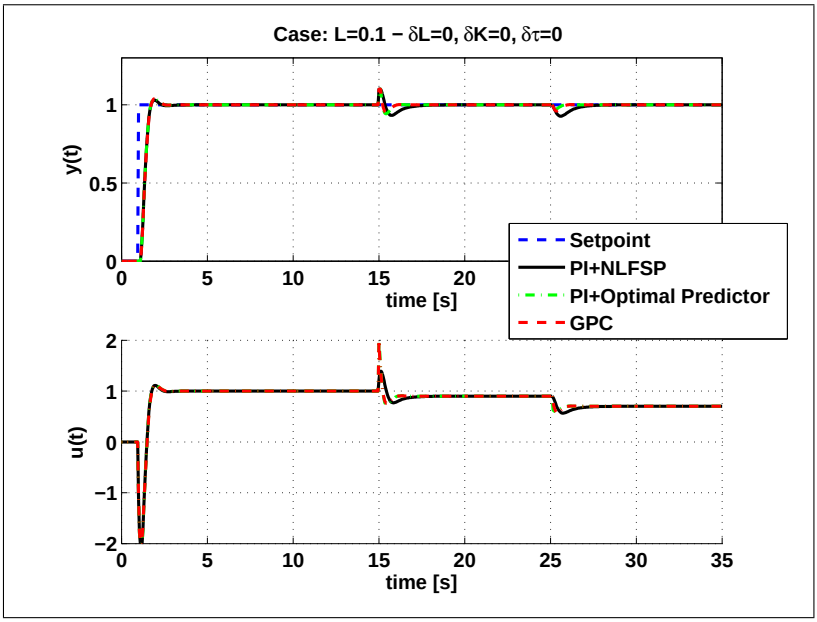

Figure 29 – Unstable Case: Output responses and control signals for the nominal system.

with unstable eigenvalues. The linear model given by

$$P(s) = \frac{(1+\delta K)}{1 - s(1+\delta\tau)} e^{-(0.1+\delta L)s}$$

with input saturation. For this example, $|u(t)| < 2$ and $T_s = 0.05$ [s]. Once more, the same controllers used in the stable case will be used here, and they also follow Assumption 5.1. The GPC control law was defined by using $N = 30$ and $Q_u = 0.5$. The 2DOF PI control law is the same as in the stable case, which is given by Eqs. (5.27)-(5.29), but the parameters of the controller are: $k_p = -7.2$, $k_i = -22.8$, $k_a = 0.01$, with a reference filter defined as in the stable case. In order to guarantee internal stability and to improve robustness with respect to the optimal predictor and the GPC, $F_r(z)$ was obtained directly in

discrete-time as follows [11]:

$$F_r(z) = \frac{0.265(z^2 - 0.966z)}{(z - 0.905)^2}.$$

The closed-loop responses for the nominal case and the case with dead-time estimation error ($\delta k = 0.1$, $\delta L = 0.1$, and $\delta\tau = 0.1$) are shown in Figures 29 and 30. Note that the NLFSP approach can be used to control open-loop unstable processes. Moreover, if the robustness filter is correctly defined, it can be designed to improve robustness margins.

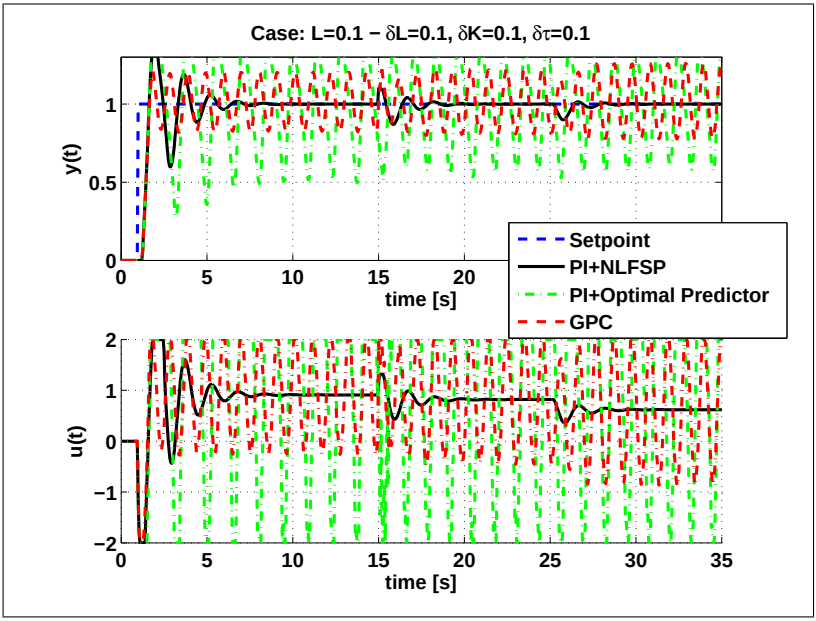

Figure 30 – Unstable Case: Output responses and control signals for the uncertain system.

For this simulation, the 2% settling time is about 2.5 [s] in the nominal case and the dead-time estimation error is 0.05 [s]. Thus, it is important to emphasize that this dead-time estimation error is considerably small if compared with the closed-loop dynamics, but output response is unstable for the optimal predictor scenario.

[11] The filter zero is computed to guarantee internal stability condition, details about robustness filter design as explained in Section 2.2.1.

An important remark is that the GPC robustness could be improved by means of a measurement filter $T(z)$, but the main advantage of the proposed approach comes from its generality since GPC and optimal predictor dead-time compensation schemes can be analyzed as a particular case of the FSP. Moreover, for simple control problems such as FOPDT, a PI controller with an anti-windup scheme can provide interesting responses with a low complexity control law. The main difficulty, however, comes from the lack of a systematic procedure for the definition of the anti-windup gain. For the linear case, the advantages of the FSP and the MPC approach can be combined in a DTC-MPC algorithm [3].

5.6 SIMULATED CASE STUDY

To illustrate the advantages of the use of the nonlinear FSP in a predictor-based nonlinear controller, this section presents a simulated case study considering a Continuous Stirred Tank Reactor (CSTR).

Reactors are very important units in chemical plants. These reactors come in many forms, but one of the most common idealization is the Continuous Stirred Tank Reactor (CSTR) [105]. These systems may exhibit highly nonlinear dynamics, especially when consecutive and side reactions are present. The process in consideration is the production of cyclopentenol (B) from cyclopentadiene (A) by acid-catalysed electrophilic addition of water in dilute solution, which is represented in Figure 31. The system also produces the side products dicyclopentadiene (D) and cyclopentanediol (C) [106]. The reaction mechanism is attributed to van de Vusse [107] and can be written as

$$\mathrm{A} \xrightarrow{k_1} \mathrm{B} \xrightarrow{k_2} \mathrm{C}$$
$$2\mathrm{A} \xrightarrow{k_3} \mathrm{D}$$

The rate constants k_i vary with temperature, but it will be considered that the reactor is operating at a constant temperature (it is isothermal), so the variables k_i are constants [105]. The equations that govern the production of B inside the reactor are [105, 106]

$$\begin{cases} \dfrac{dC_a(t)}{dt} = -k_1 C_a(t) - k_3 C_a(t)^2 + (C_{af}(t) - C_a(t))\dfrac{F(t)}{V} & \text{(5.30a)} \\ \dfrac{dC_b(t)}{dt} = k_1 C_a(t) - k_2 C_b(t) - C_b(t)\dfrac{F(t)}{V} & \text{(5.30b)} \end{cases}$$

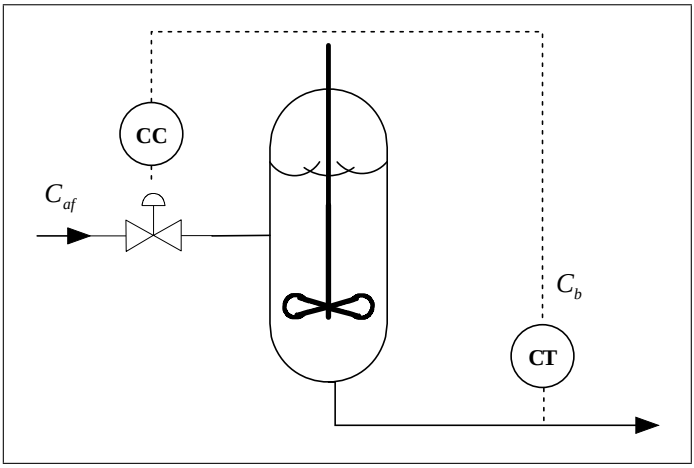

Figure 31 – The CSTR process with an analyzer.

where the desired output C_b [mol/l] is the concentration of product B, C_a and C_{af} are the concentrations of A [mol/l] in the reactor and in the feed, respectively. The manipulated input is the dilution rate F [l/min], V is the constant reactor volume [l], and the rate constants are given by $k_1 = k_2 = 0.8433$ [min^{-1}], $k_3 = 0.1123$ [mol/(l min)] [106].

The analyzer that measures the concentration C_b is located at a distance l from the reactor. This adds a dead time between the real concentration inside the reactor and the measured value. The delay is $d_0 = d_t + d_a$, where d_a is the time taken by the analyzer to give the value of the measured concentration and d_t is the effective transport dead time because of the distance l. Thus, the model of the dead-time nonlinear process has two parts: the nonlinear dynamics of the reactions and the measurement dead time. The dead time will be considered to be variable between 0.36 and 1.08 minutes due to the flow and analysis time variations [3].

The operating point was chosen considering the standard operation of the CSTR [106]: $\overline{C_{af}} = 5.1$, $\overline{C_a} = 1.235$, $\overline{C_b} = 0.9$ (all the concentrations are in [mol/l]) and $\overline{F}/V = 0.3138$ [min^{-1}] (henceforth, the input of the system will be given normalized in relation to the reactor volume). In Figure 32 the static values of C_b and C_a for different values of F/V (and a fixed value of C_{af}) are shown. It is clear that the system is highly nonlinear, even when considering only the static gain.

A system identification was performed, resulting in a sixth order

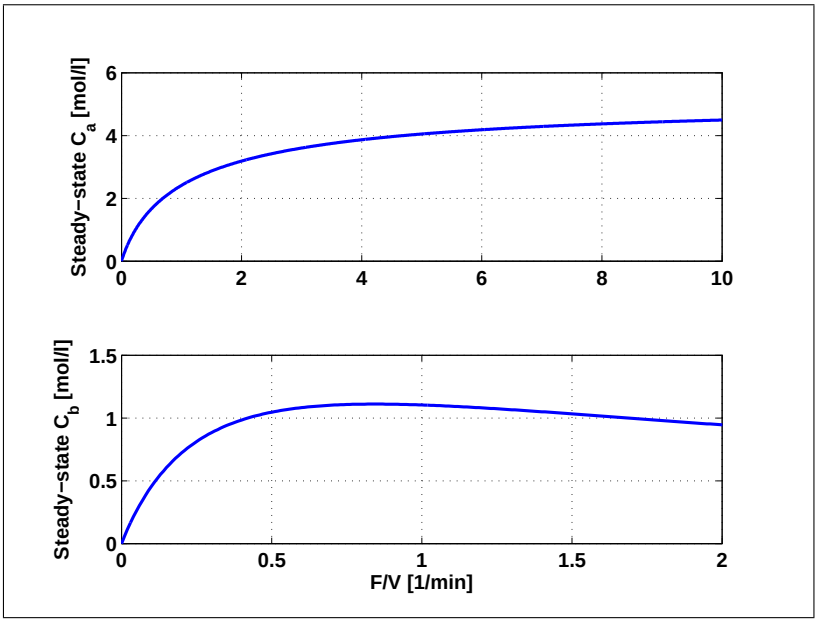

Figure 32 – Static characteristics of the CSTR.

Volterra model (Eq. (5.1)) where polynomials $A(q)$ and $B(q)$ have order 2 and 1, respectively. The model can be represented in state-space form as

$$\mathbf{x}(t+1) = \mathbf{A}\mathbf{x}(t) + g(\mathbf{u}(t)) + \mathbf{w}(t) \tag{5.31}$$

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}(t-d) \tag{5.32}$$

with $g(.) = [g_1(.), 0]^T$, and $g_1(.) : \mathbb{R}^2 \to \mathbb{R}$ defined as

$$\begin{aligned} g_1(\mathbf{u}(t)) =& b_0 u(t) + b_1 u(t-1) + h_1 u(t)^2 + h_2 u(t)^3 \\ &+ h_3 u(t)^4 + h_4 u(t)^5 + h_5 u(t) u(t-1) \\ &+ h_6 u(t)^2 u(t-1)^2 + h_7 u(t)^3 u(t-1)^3, \end{aligned}$$

and

$$\mathbf{A} = \begin{bmatrix} a_1 & a_2 \\ 1 & 0 \end{bmatrix}.$$

The states are $\mathbf{x}(t) = [C_b(t), C_b(t-1)]$, which are all measurable since they are simply the delayed output. The input is $\mathbf{u}(t) = [F(t)/V, F(t-1)/V]$, and the disturbance is $\mathbf{w}(t) = [w_1(t), 0]^T$. The

disturbance affects directly only the first state since the others are the delayed outputs. The values of the coefficients present in the system equations are: $a_1 = 1.3664$, $a_2 = -0.4311$, $b_0 = -0.1053$, $b_1 = 0.4530$, $h_1 = -0.0134$, $h_2 = 0.8461$, $h_3 = -0.7602$, $h_4 = 0.1885$, $h_5 = -0.7191$, $h_6 = 0.2073$ and $h_7 = -0.026$. The sampling time was chosen as $T_s = 0.18$ [min]. The nominal dead time is considered $d_0 = 0.72$ [min], which results in a discrete dead time of $d = 4$ samples. In Figures 33 and 34 the model and the process are compared, in the first one the static characteristics are presented, and in the second the process and the system's transient responses are shown. From the results, it can be seen that the model manages to represent well the static and transient characteristics of the CSTR for the input range $[0.1, 2]$ $[\text{min}^{-1}]$.

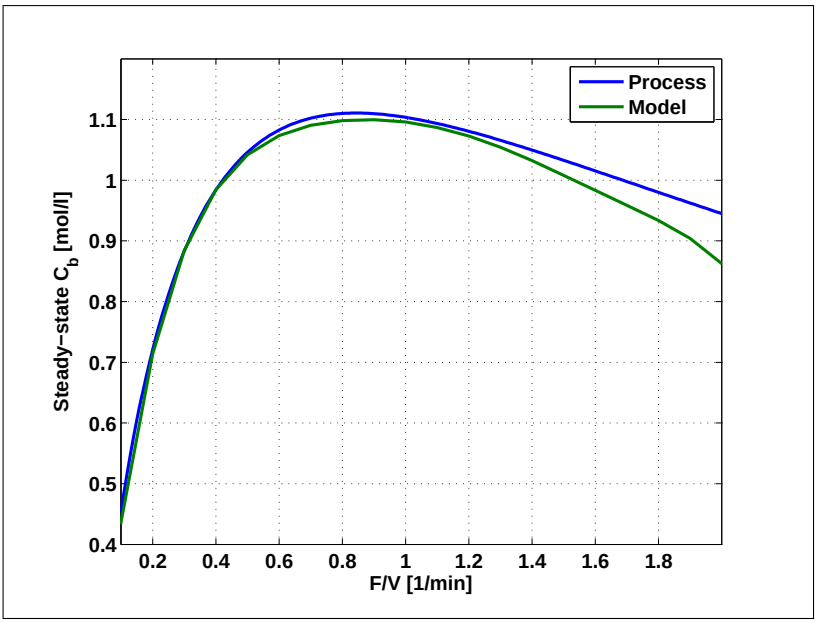

Figure 33 – Comparison of the steady state values of C_b of the process and the model.

Since the analysis of the optimal predictor and the NLFSP was done with the system definition Eq. (5.4) where the delay is on the input, a variable change is used to obtain the CSTR model described by Eqs. (5.31) and (5.32) in the proper form. Defining

$$\boldsymbol{\xi}(t) = \mathbf{x}(t - d),$$

the following system with input delay is obtained

$$\begin{aligned}\boldsymbol{\xi}(t+1) =& \mathbf{A}\boldsymbol{\xi}(t) + g(\mathbf{u}(t-d)) + \mathbf{w}(t-d)\\ y(t) =& \begin{bmatrix} 1 & 0 \end{bmatrix} \boldsymbol{\xi}(t).\end{aligned}$$

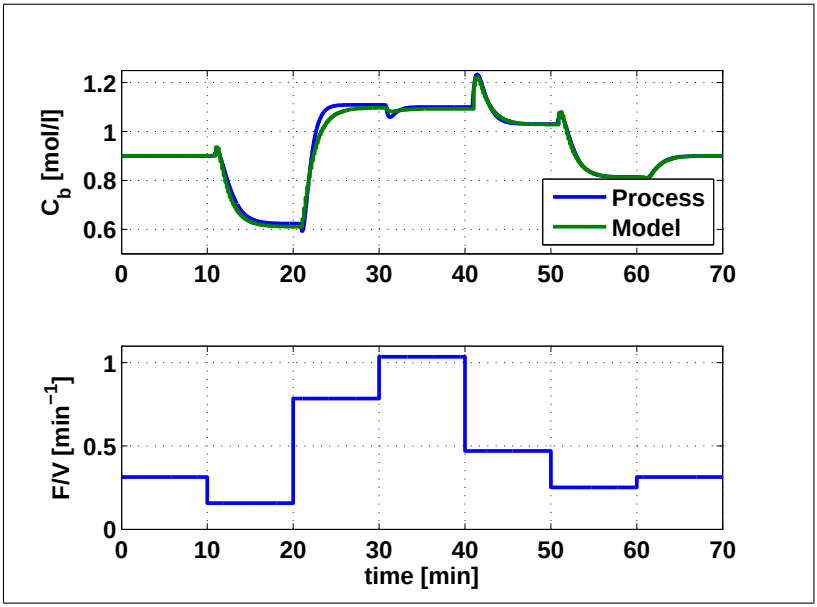

Figure 34 – Comparison of the steady state values of C_b of the process and the model.

5.6.1 Control of the CSTR

The control of this process was made using the nonlinear MPC algorithm PNMPC, which was described in section 3.3.4 [82, 83]. Apart from the nonlinear model, the tuning parameters of this algorithm are essentially the same of other largely used MPC controllers, e.g., DMC and GPC [2, 3]. The simulation results will consider the original PNMPC, which uses the optimal predictor described earlier, and the PNMPC with the proposed NLFSP. In all cases, Assumption 5.1 holds.

Following the specifications in [106], the closed-loop system must reach steady-state value in less than 20 minutes for set-point changes

with a well-damped smooth transient response. To achieve the specifications, the PNMPC was tuned with an initial prediction horizon $N_1 = 5$ (to avoid the dead time) and $N_2 = 104$ as the final horizon. The control horizon is $N_u = 30$ and the weights of the error and of the control increments are, respectively, $Q_y = 1$ and $Q_u = 30$. For the PNMPC-NLFSP, only the prediction horizons change, since the PNMPC will control the equivalent dead-time-free model, hence, $N_1 = 1$ and $N_2 = 100$. Also, for the NLFSP case, a filter was designed with the method explained in Section 5.4.4, to cancel the slow open-loop poles of the system, which are present in the disturbance rejection response. The open-loop poles are the eigenvalues of $\mathbf{A}$, which are: 0.8720 and 0.4944. The filter will be tuned to remove only the slower pole (0.8720), which has a greater influence on the closed-loop response. According to Section 2.2.1, considering the prediction filter $\mathbf{F_{r1}}(z) = F_{r1}(z)\mathbf{I}$, the condition that the filter must satisfy in order to cancel the undesirable open-loop pole is

$$(1 - z^{-d}F_{r1}(z))\big|_{z=0.8720} = 0$$

which results in the following filter[12]

$$F_{r1}(z) = \frac{1.3280(z - 0.8118)}{(z - 0.5)^2},$$

where the poles of the filter (0.5) were chosen to obtain a faster response than the open-loop one.

To verify if the tuning parameters were chosen adequately, simulations were made with the original PNPMC algorithm, which uses the optimal predictor, and the PNMPC-NLFSP with the filter $\mathbf{F_{r1}}(z)$, tuned to improve the disturbance rejection. The simulation results are shown in Figure 35. Initially the system is at steady-state, then, the set-point is changed from 0.9 to 0.8 [mol/l] at time 2 [min], and then from 0.8 to 1 [mol/l] at time 18 [min]. At 36 [min] an input disturbance is added with an amplitude of 0.1 [min^{-1}]. The transients of the set-point changes, in all cases, are smooth and achieve steady-state in approximately 10 [min] for both controllers. This is expected since the simulations were done using the nominal model, i.e., there are no disturbances because the model is perfect, hence, in this case, the predictor will not influence the response. However, this changes when the unmea-

[12]This filter has one excess pole to help attenuate the magnitude of the filter at high frequencies, which will contribute to robustness and noise attenuation of the closed-loop system.

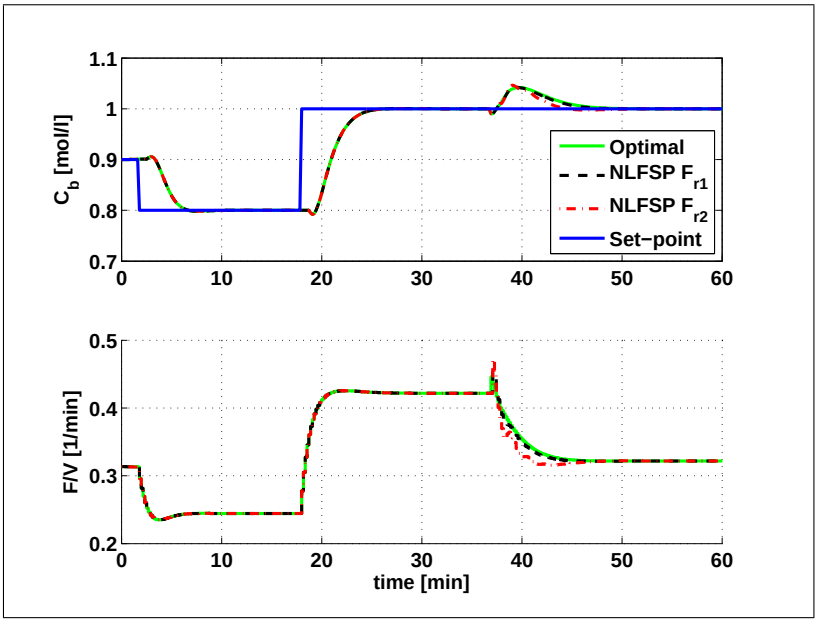

Figure 35 – Nominal closed-loop response of the CSTR with the PNMPC controller and the different predictors.

surable input disturbance is added. The disturbance rejection is shown more clearly in Figure 36. With the optimal predictor, the disturbance is rejected in 7.52 [min], while in the NLFSP with $\mathbf{F_{r1}}(z)$ the rejection occurs after 6.62 [min], a 12 % improvement. The improvement is not bigger because the disturbance rejection is a combination of the open-loop dynamics and of the closed-loop dynamics of the dead-time-free system, as discussed in Section 5.4.3. Since the slow open-loop pole ($z = 0.8742$) was canceled with the prediction filter, what dominates the disturbance rejection response is the closed-loop dynamics of the dead-time-free system. Hence, a fine-tuning of the prediction filter was required. After some simulation tests, the following filter was obtained, which improves the disturbance rejection in 45 % (from 7.52 [min] to 4.10 [min]),

$$\mathbf{F_{r2}}(z) = F_{r2}(z)\mathbf{I} = \frac{2.1686(z - 0.8847)}{(z - 0.5)^2}.$$

From the magnitude diagrams of the filters (Figure 37), note that

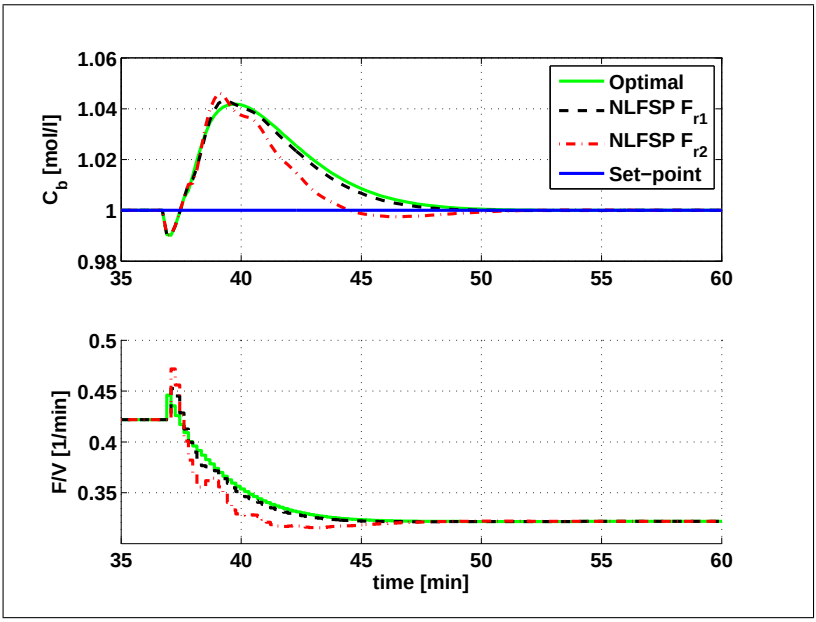

Figure 36 – Disturbance rejection response CSTR for the nominal case.

$\mathbf{F_{r2}}(z)$ has a greater amplification at medium and high frequencies than $\mathbf{F_{r1}}(z)$. The set-point tracking and disturbance rejection responses are shown in Figures 35 and 36.

Also, note that instead of using the prediction filter, the disturbance rejection response could also be improved reducing the weights of the control increments in the PNMPC cost function. This was not done because this would change the primary controller, and the focus of this thesis is to study how different predictor structures influence the robustness and closed-loop responses with a fixed primary controller. As the filters $\mathbf{F_{r1}}(z)$ and $\mathbf{F_{r2}}(z)$ have high-pass characteristics (see Figure 37), they will have a negative impact on the closed-loop robustness, hence, it must be reinforced that the use of the prediction filter to improve the disturbance rejection response is only practical if the nominal model is good, i.e., the disturbance due to modeling errors is negligible.

Finally, the PNMPC controller with the different predictor structure is used to control the the process simulated using the differential equations of the CSTR given by Eq. (5.30). The simulation, in this

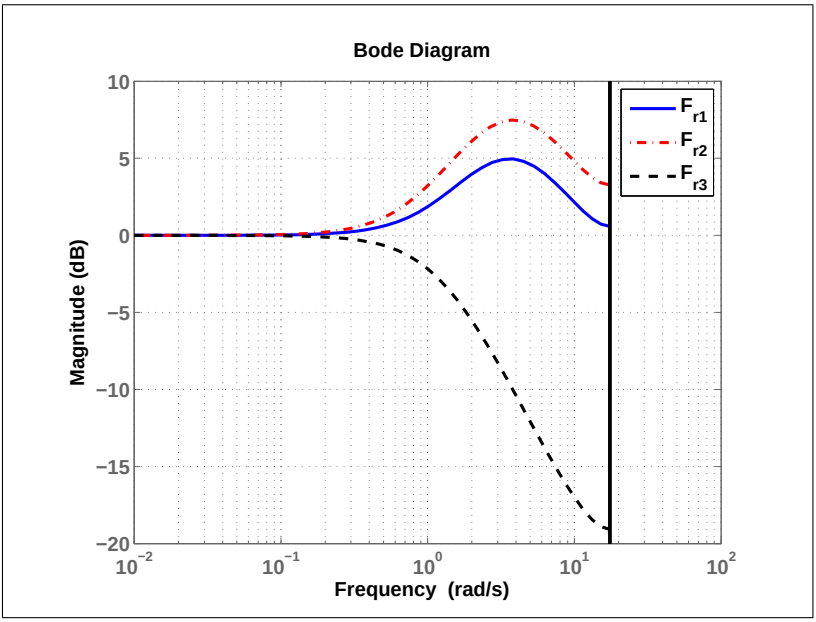

Figure 37 – Magnitude diagram of the prediction filters $\mathbf{F_{r1}}(z)$, $\mathbf{F_{r2}}(z)$ and $\mathbf{F_{r3}}(z)$.

case, is similar to the nominal one, initially the system is at steady-state, than, the set-point is changed from 0.9 to 0.8 [mol/l] at time 2 [min], and then from 0.8 to 1 [mol/l] at time 18 [min]. At 36 [min] an input disturbance is added with an amplitude of 0.1 [min^{-1}], and, at time $t = 38$ [min] a +10 % variation of the feed concentration C_{af} is also included. After time $t = 70$ [min], an additive zero-mean white noise with a standard deviation of 0.005 was added to all simulations. The noise is not added during the entire simulation to make it easier to distinguish the effects of the noise, process' disturbance and model mismatch on the value of the disturbance $\mathbf{w}$.

The simulation results are presented in Figure 38. The transients of the set-point changes are smooth and achieve steady-state in approximately 10 minutes in all cases, as expected from the nominal system response. The interesting results lie in the analysis of the disturbance $\mathbf{w}$ and the equivalent disturbance $\widetilde{\mathbf{w}}$ shown in Figure 39. The disturbances are shown separately because the values of $\mathbf{w}(t)$ are partially dependent on the control input, hence, for each controller, there is

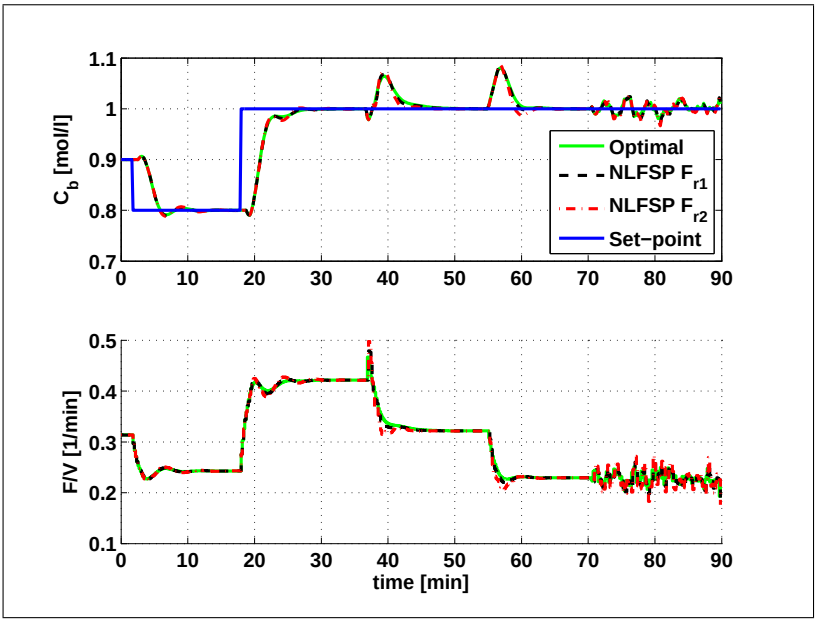

Figure 38 – Closed-loop response of the CSTR using its differential equations.

a different disturbance set. The first plot shown in Figure 39 displays the disturbances for the PNMPC with the optimal predictor and the second and third plots the disturbances for the PNMPC-NLFSP with $\mathbf{F_{r1}}(z)$ and $\mathbf{F_{r2}}(z)$, respectively. In all cases, the simulations are consistent with the theoretical results, i.e., the bound on the equivalent disturbance $\widetilde{\mathbf{w}}$ is bigger than the bound on the original disturbance $\mathbf{w}$. For the optimal predictor, this was proven through Theorem 5.1, and for the NLFSP case, note that the filters have high-pass behaviour, resulting in a nonmonotonic behaviour, hence, the sum of their impulse-response coefficients will be greater than 1, thus increasing the equivalent disturbance $\widetilde{\mathbf{w}}$.

For the optimal predictor case, $|\mathbf{w}| \leq \gamma = 0.0293$ and, considering the euclidian norm, $|\mathbf{A}'| \cong 8.0$. With this information it is possible to compute the limit of the set of $\widetilde{\mathbf{w}}$

$$|\widetilde{\mathbf{w}}| \leq (|\mathbf{I} + \mathbf{A}'| + |\mathbf{A}'|)\gamma \cong 16.5223\gamma = 0.4841$$

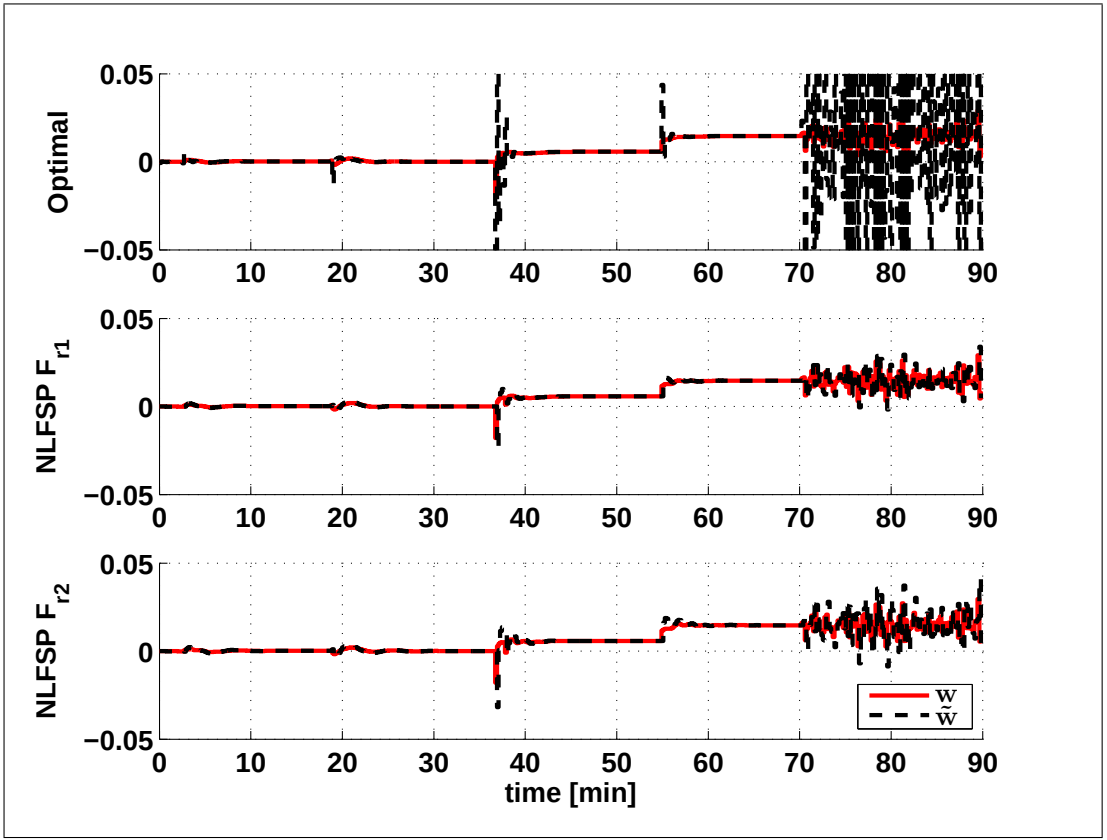

Figure 39 – Disturbances of the closed-loop CSTR.

Using the simulation data, $|\widetilde{\mathbf{w}}| \leq 0.1724$, which lies inside the upper bound obtained previously, which is expected because of the conservativeness of Eq. (5.12). In spite of this, even considering the bound 0.1724, this value is 5.9 times bigger than γ. This result shows that the optimal predictor increases the disturbance the controller perceives. This behaviour is especially apparent in the presence of noise, which is introduced after $t = 70$ [min].

For the PNMPC-NLFSP with $\mathbf{F_{r1}}(z)$, $|\mathbf{w}| \leq 0.0291$ and $|\widetilde{\mathbf{w}}| \leq 0.0339$ (1.2 times bigger), and with $\mathbf{F_{r2}}(z)$, $|\mathbf{w}| \leq 0.0296$ and $|\widetilde{\mathbf{w}}| \leq 0.0414$ (1.4 times bigger). They also increase the original disturbance because they have a high-pass behaviour, which is necessary to improve the disturbance rejection. An important observation is that both filters manage to obtain a better disturbance rejection than the optimal predictor, while maintaining the bound on $\widetilde{\mathbf{w}}$ smaller than in the optimal predictor case. This happens because the filters use the disturbance $\mathbf{w}$ information in a more efficient way.

Now, consider the case where there is a dead-time mismatch. Instead of $d = 4$ samples, the process exhibits a dead time of $d = 6$ samples. This will incorporate more modelling errors in the disturbance $\mathbf{w}$. The closed-loop responses of the PNMPC-NLFSP with filters $\mathbf{F_{r1}}(z)$ and $\mathbf{F_{r2}}(z)$ are shown in Figure 40. It is clear that the system is unstable if any of these filters are used. This happened because the dead-time mismatch introduced medium and high-frequency disturbances, which are exactly the ones the filters amplify. Hence, if there is dead-time uncertainty, the controller designer must take into account that the improvement of the disturbance rejection will have a great impact on the robustness of the closed-loop system.

The simulation results for the PNMPC with optimal predictor is shown in Figures 41 and 42. Even with a mismatch of 2 samples, the PNMPC with optimal predictor exhibits a high-frequency oscillatory behaviour that degenerates the system response, especially the system input, where the amplitude of the oscillations is bigger. The reason for this becomes clear with the analysis of the disturbances, which are shown in the first plot of Figure 42. In this simulation case, $|\mathbf{w}| \leq \gamma = 0.0344$, and $|\widetilde{\mathbf{w}}| \leq 0.2040$ (obtained analysing the simulation data), which is 6.0 times bigger.

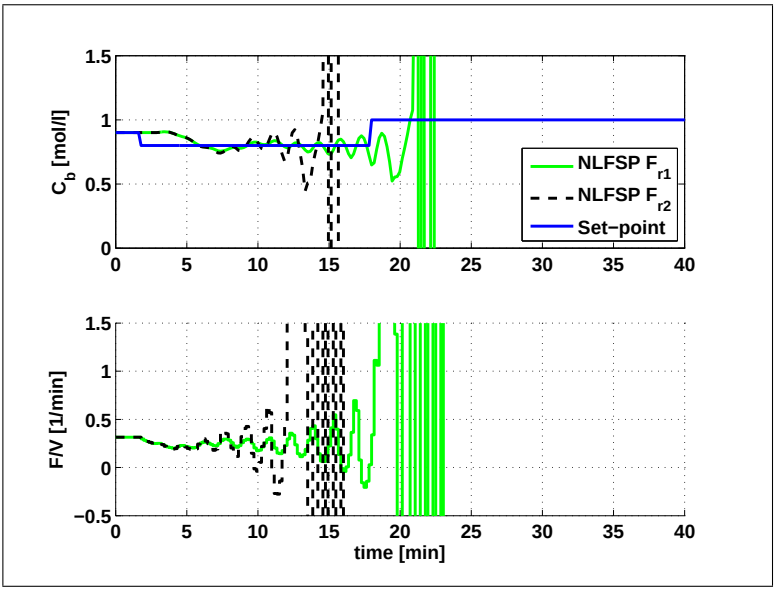

Figure 40 – Control of the CSTR using the prediction filters $\mathbf{F_{r1}}(z)$ and $\mathbf{F_{r2}}(z)$ with dead-time mismatch.

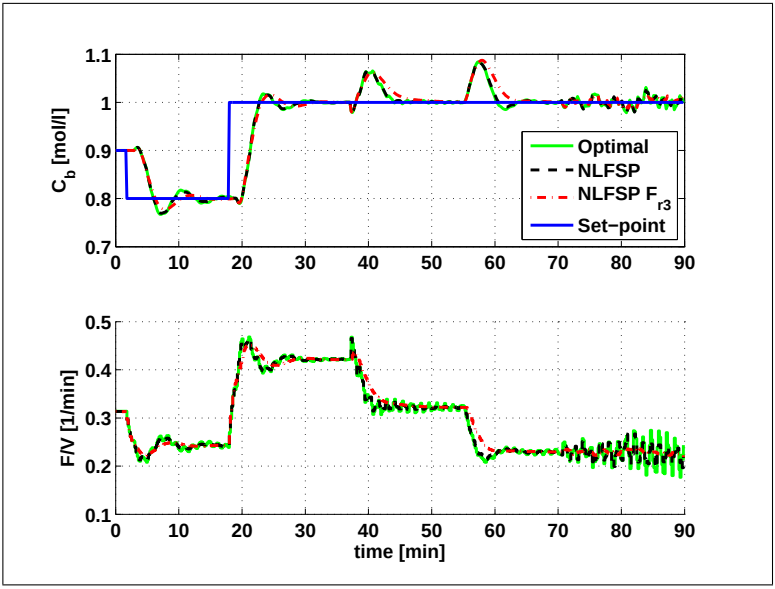

Figure 41 – Closed-loop response of the CSTR with dead-time mismatch.

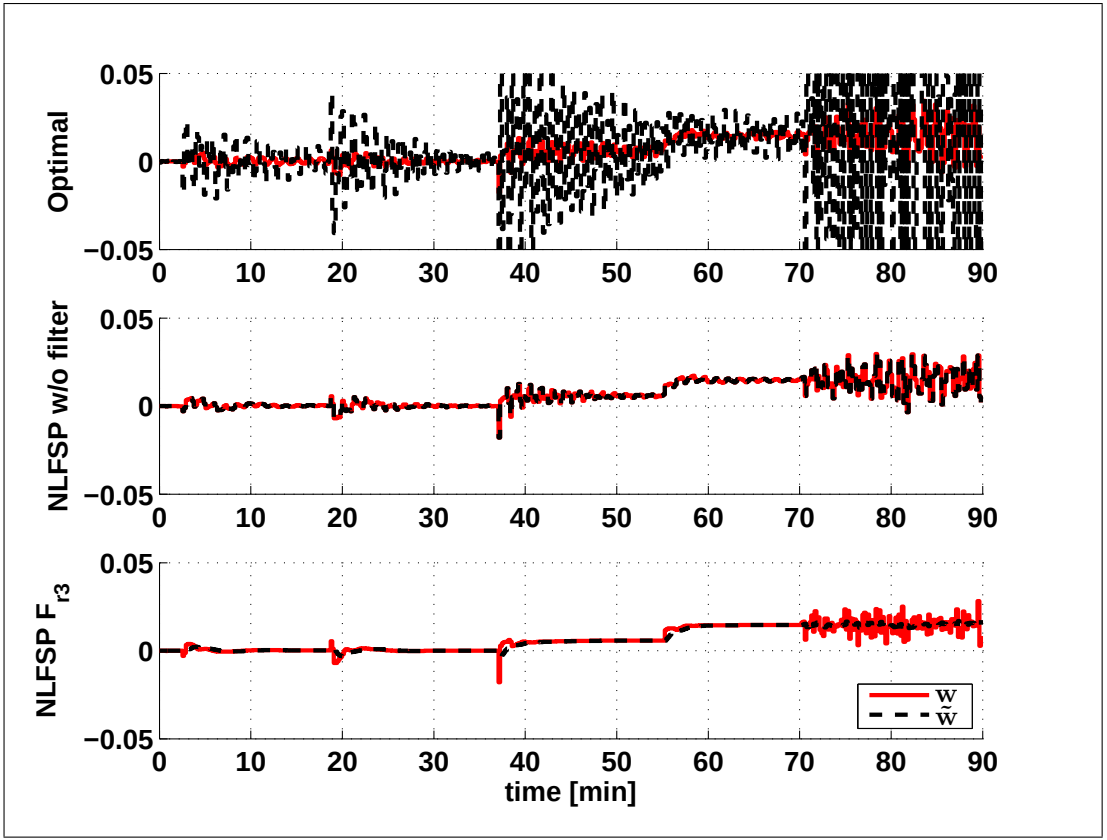

Figure 42 – Disturbances of the closed-loop CSTR with dead-time mismatch.

To improve the robustness of the closed-loop system, the PNMPC-NLFSP without filter, i.e., $\mathbf{F_r}(z) = \mathbf{I}$, was tested. The results are shown in Figures 41 and 42.

This disturbance amplification does not happen with the PNMPC-NLFSP (Figure 42), which can be seen by the ouput/input response of the system (Figure 41), which, again, confirms the theoretical results (Theorem 5.2). The PNMPC-NLFSP output response is smoother than with the optimal predictor, especially during the disturbance rejection, but the input still has some oscillatory behaviour, albeit with a smaller amplitude. To avoid this problem, tuning procedure presented in Section 5.4.5 will be used. Analysing the simulation data, the frequency characteristics of $\widetilde{\mathbf{w}}$ were obtained using the Fast Fourier Transform algorithm, and they are presented in Figure 43.

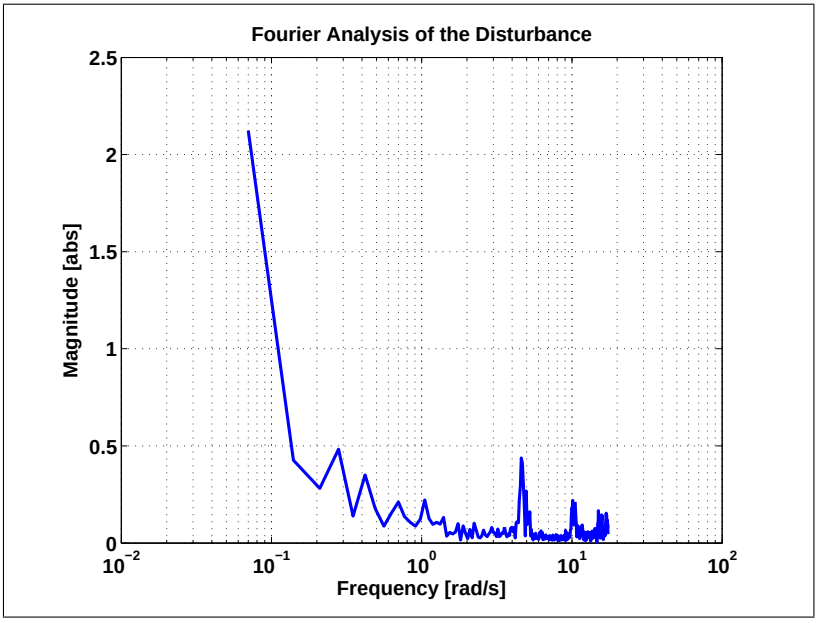

Figure 43 – Fourier analysis of $\mathbf{w}$ in the PNMPC-NLFSP case with $\mathbf{F_r}(z) = \mathbf{I}$.

The results of this analysis show that $\mathbf{w}$ has significant frequency components near 5 [rad/s]. Hence, the following low-pass filter was

designed to attenuate these components:

$$\mathbf{F_{r3}}(z) = F_{r3}(z)\mathbf{I} = \frac{0.2}{z - 0.8}\mathbf{I},$$

and its frequency characteristics are shown in Figure 37.

The addition of the filter $\mathbf{F_{r3}}(z)$ results in a smooth response for both the input and output, as can be seen in Figure 41. From the analysis of the disturbance data in this case (Figure 42), $|\mathbf{w}| \leq \gamma = 0.0282$, and $|\widetilde{\mathbf{w}}| \leq 0.0178 \leq \gamma$, i.e., the bound on $\widetilde{\mathbf{w}}$ is smaller than the bound on $\mathbf{w}$. This happens because the bound on $\mathbf{w}$ must account for the peak values of $\mathbf{w}$, but, as $\widetilde{\mathbf{w}}$ is $\mathbf{w}$ filtered by a low-pass filter, the peaks are cut off, reducing the bound on $\widetilde{\mathbf{w}}$. These peaks come mostly from the model mismatch, as can be seen in the simulation results up to $t = 38$ [min], before the insertion of the process disturbances and noise.

5.7 FINAL COMMENTS

In this chapter a Nonlinear Filtered Smith Predictor (NLFSP) for systems with input nonlinearities was proposed following the ideas of the linear FSP. Input-saturation, Volterra models and Hammerstein models can be treated by using the proposed technique. It was shown that this new predictor, if correctly defined, can be used to improve the disturbance rejection response or the closed-loop robustness if compared with optimal predictors. Optimal predictors are important since they are implicitly used in many MPC algorithms, however, robustness behavior is indirectly defined by the implicit disturbance model which may result in poor robustness margin depending on the nominal dead time. Hence, substituting the implicit optimal predictor by an explicit NLFSP can be advantageous, as shown in Sections 5.5 and 5.6.

Moreover, the results presented in this chapter can be analyzed as an extension of the Filtered Smith Predictor (FSP) properties for systems with input nonlinearities. This is an important issue since robustness margins for open-loop stable systems do not depend on the nominal dead time length for the FSP strategy. It is also important to note that although the NLFSP was presented in the MPC context, it can be used with any controller.

6 FILTERED SMITH PREDICTOR FOR STABLE NONLINEAR SYSTEMS

This chapter extends the results presented in Chapter 5 to the more general case of stable nonlinear systems. As stated before for processes with input nonlinearities, the NLFSP provides the following characteristics: (i) the nominal set-point performance is not affected by the prediction filter, (ii) it is possible to adjust the disturbance rejection response, (iii) robustness and noise attenuation can be improved by a suitable tuning of the predictor filter, and (iv) it can also be applied to unstable processes. However, only items (i) and (iii) will be proven here for the general stable nonlinear case.

The rest of the chapter is organized as follows: in Section 6.1 the general nonlinear system description will be presented, and in Section 6.2.1 the NLFSP will be analyzed. As will be seen in Section 6.2.1, this system description is inadequate to proof the properties of the NLFSP, hence, an alternative nonlinear system description will be presented in Section 6.3, then the optimal and the NLFSP prediction schemes will be discussed in Sections 6.4 and 6.5, respectively. Also, an illustrative example and a case study are presented in Sections 6.6 and 6.7, respectively.

6.1 SYSTEM DESCRIPTION

This chapter considers the control of processes that can be modeled by a nonlinear MIMO uncertain time-invariant stable discrete time system given by

$$\begin{bmatrix} \mathbf{x_1}(t+1) \\ \vdots \\ \mathbf{x_p}(t+1) \end{bmatrix} = \begin{bmatrix} f_1(\mathbf{x_1}(t), \mathbf{u}(t-d_1)) \\ \vdots \\ f_p(\mathbf{x_p}(t), \mathbf{u}(t-d_p)) \end{bmatrix} + \begin{bmatrix} \overline{\mathbf{w}}_\mathbf{1}(t) \\ \vdots \\ \overline{\mathbf{w}}_\mathbf{p}(t) \end{bmatrix} \tag{6.1}$$

or, in a simplified manner, as

$$\mathbf{x}(t+1) = f(\mathbf{x}(t), \mathbf{u}(t-D)) + \overline{\mathbf{w}}(t). \tag{6.2}$$

The states vector $\mathbf{x}$ is separated in p groups in a way that it is possible to distinguish the minimum dead times from the input vector $\mathbf{u}$ to each group of states. Hence, $\mathbf{x_i} \in \mathbb{R}^{n_i}$, $\mathbf{x} \in \mathbb{R}^n$, with $n = \sum_i n_i$,

and the input vector is composed of m different inputs such that $\mathbf{u}(t) = [\mathbf{u_1}(t), \mathbf{u_2}(t), \ldots, \mathbf{u_m}(t)]^T$, with $\mathbf{u_i}(t) = [u_i(t), u_i(t-1), \ldots, u_i(t-l+1)]^T$, where l defines the necessary number of past inputs[1], thus, $\mathbf{u_i} \in \mathbb{R}^l$ and $\mathbf{u} \in \mathbb{R}^{m \times l}$. $f_i(.)$ is a nonlinear continuous function[2] such that $f_i : \mathbb{R}^{n_i} \times \mathbb{R}^{m \times l} \to \mathbb{R}^{n_i}$, and the variable D represents the minimal dead time set from each group of states, $D = \{d_1, d_2, \ldots, d_p\}$, in a way that

$$f(\mathbf{x}(t-D), \mathbf{u}(t-D)) \triangleq \begin{bmatrix} f_1(\mathbf{x_1}(t-d_1), \mathbf{u}(t-d_1)) \\ \vdots \\ f_p(\mathbf{x_p}(t-d_p), \mathbf{u}(t-d_p)) \end{bmatrix}.$$

Also, $\overline{\mathbf{w}}(t) = [\overline{\mathbf{w}}_\mathbf{1}(t), \ldots, \overline{\mathbf{w}}_\mathbf{p}(t)]^T$, with $\overline{\mathbf{w}}_\mathbf{i} \in \mathbb{R}^{n_i}$ and $\overline{\mathbf{w}} \in \mathbb{R}^n$, is the unmeasurable additive disturbance.

Notice that $\overline{\mathbf{w}}(t)$ can describe any kind of mismatch between the measured state at time $t+1$ and its expected value at time t (this will be better discussed in Section 6.3.3), including model mismatch and unmeasurable process disturbances [6]. Its value can only be known at the time instant $t+1$, where $\overline{\mathbf{w}}(t)$ is computed as

$$\overline{\mathbf{w}}(t) = \mathbf{x}(t+1) - f(\mathbf{x}(t), \mathbf{u}(t-D)). \tag{6.3}$$

The nonlinear the model also follows Assumptions 6.1 and 6.2, and this last one requires the following definition:

Definition 6.1. *A function $\lambda : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$, where $\mathbb{R}_{\geq 0}$ denotes the non-negative real numbers, is of class $\mathcal{K}$ (or a "$\mathcal{K}$-function") if it is continuous, strictly increasing and $\lambda(0) = 0$.*

If a function λ is of class $\mathcal{K}$ and $\lambda(s) \to +\infty$ as $s \to +\infty$, then λ is of class $\mathcal{K}_\infty$.

Assumption 6.1. *If the model function $f(\mathbf{x}, \mathbf{u})$ is such that $f(0,0) = 0$ and it is assumed to be uniformly continuous on $\mathbf{x}$ in the domain $\mathbb{X} \times \mathbb{U}$, then there is a $\mathcal{K}$-function σ_x such that*

$$|f(\mathbf{x}_1, \mathbf{u}) - f(\mathbf{x}_2, \mathbf{u})| \leq \sigma_x(|\mathbf{x}_1 - \mathbf{x}_2|),$$

[1] The variable l can be defined individually by input and by group of states, but, for simplicity, a fixed l is considered.

[2] Note that the parameters of each function f_i are the input vector $\mathbf{u}$ and the ith group of states only. This is important to be able do distinguish multiple dead times. Also, an interesting fact is that, if this system were to be linearized in a state-space form, the resulting matrix $\mathbf{A}$ would be block diagonal, similarly to the system with input nonlinearities given by Eq. (5.4).

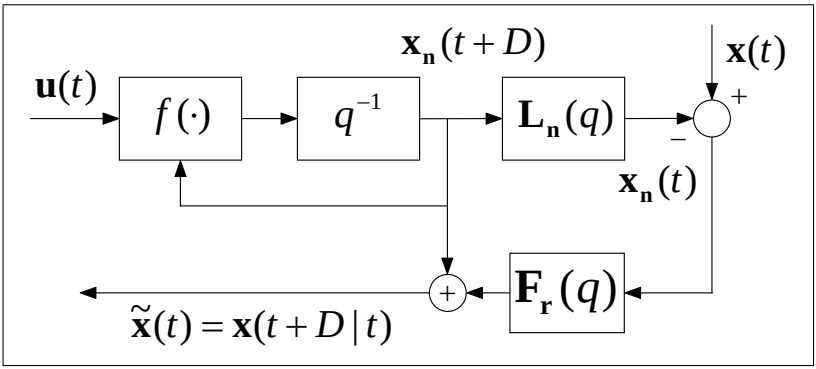

Figure 44 – Block diagram of the NLFSP.

for all $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{X}$*, and for all* $\mathbf{u} \in \mathbb{U}$*.*

Assumption 6.2. *The nonlinear system only has stable equilibrium points in the domain being considered, i.e., if* $\mathbf{x} \in \mathbb{X}$ *and* $\mathbf{u} \in \mathbb{U}$*, with* $\mathbb{X}, \mathbb{U} \subset \mathbb{R}$*, the linearized model is open-loop stable for any equilibrium point in the domain* $\mathbb{X} \times \mathbb{U}$*.*

As in the case of systems with input nonlinearities, the concept of local ISS can also be applied to this system, hence, the bound on $\overline{\mathbf{w}}(t)$ plays an important role in the robustness analysis of the predictor structures presented in this work. The bigger the bound on $\overline{\mathbf{w}}(t)$, the harder it is to guarantee stability.

Additionally, the following dead-time-free model is used to analyze the predictors:

$$\widetilde{\mathbf{x}}(t+1) = f(\widetilde{\mathbf{x}}(t), \mathbf{u}(t)) + \widetilde{\mathbf{w}}(t), \tag{6.4}$$

where $\widetilde{\mathbf{x}}(t) \triangleq \mathbf{x}(t+D|t)$, in a way that $\widetilde{\mathbf{x}}(t+1) = \mathbf{x}(t+D+1|t+1)$, as defined in Eq. (5.9).

6.2 NLFSP FOR NONLINEAR SYSTEMS

The equations of the NLFSP for nonlinear systems described in Section 6.1 are given by

$$\begin{cases} \mathbf{x_n}(t+D) = f(\mathbf{x_n}(t+D-1), \mathbf{u}(t-1)) & \text{(6.5a)} \\ \mathbf{x}(t+D|t) = \mathbf{x_n}(t+D) + \mathbf{F_r}(q)(\mathbf{x}(t) - \mathbf{x_n}(t)), & \text{(6.5b)} \end{cases}$$

where $\mathbf{x_n}$ represents the nominal evolution of the system states, i.e., without taking into account disturbances, $\mathbf{F_r}(q) = F_r(q)\mathbf{I}$, $\mathbf{I}$ is an identity matrix and $F_r(q)$ is a stable SISO filter in the delay operator q with unitary static gain[3], i.e., $F_r(1) = 1$. The block diagram representation of the NLFSP is shown in Figure 44.

6.2.1 Analysis of the NLFSP

As was done in Chapter 5, the robustness of the predictor structures are analyzed through the bound on the equivalent disturbance $\widetilde{\mathbf{w}}$.

In the case of the NLFSP, the following theorem is needed:

Theorem 6.1. *Consider an open-loop stable nonlinear dead-time system (Assumption 6.2) described by Eq. (6.2), and the dead-time-free model given by Eq. (6.4), where the function $f(.)$ follows Assumption 6.1. If the NLFSP, which is described by Eq. (6.5), is used, the bound on $\widetilde{\mathbf{w}}$ at time t is*

$$\begin{aligned}|\widetilde{\mathbf{w}}(t)| \leq & \sigma_x(|\mathbf{F_r}(q)(\mathbf{x}(t) - \mathbf{x_n}(t))|) \\ & + |\mathbf{F_r}(q)\left[f(\mathbf{x}(t), \mathbf{u}(t-D)) - f(\mathbf{x_n}(t), \mathbf{u}(t-D))\right]| \\ & + |\mathbf{F_r}(q)\overline{\mathbf{w}}(t)|. \end{aligned} \tag{6.6}$$

Proof. See Appendix A.3. □

From Theorem 6.1, it is not possible to affirm that the equivalent disturbance $\widetilde{\mathbf{w}}$ is simply the original disturbance $\overline{\mathbf{w}}$ filtered by $\mathbf{F_r}(q)$, as was done in the case of systems with input nonlinearities (see Theorem 5.2). Hence, the bound on $\widetilde{\mathbf{w}}$ can not be obtained without being conservative.

To simplify the analysis of the bound on $\widetilde{\mathbf{w}}$, consider that $\mathbf{F_r}(q) = \mathbf{I}$, and that the states and $\overline{\mathbf{w}}$ are bounded, i.e., $\mathbf{x}, \mathbf{x_n} \in \mathbb{X} = \{\mathbf{a} \in \mathbb{R}^n :$

[3]The prediction filter can also be block diagonal

$$\mathbf{F_r}(q) = \begin{bmatrix} \mathbf{F_{r1}}(q) & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \mathbf{F_{rp}}(q) \end{bmatrix},$$

with $\mathbf{F_{ri}}(q) = F_{ri}(q)\mathbf{I}$, where $\mathbf{F_{ri}}(q)$ has the dimension of the ith group of states, i.e., each group of states can have an independent prediction filter.

$|\mathbf{a}| \leq \gamma_x\}$, and $\overline{\mathbf{w}} \in \overline{\mathbb{W}} = \{\overline{\mathbf{w}} \in \mathbb{R}^n : |\overline{\mathbf{w}}| \leq \overline{\gamma}\}$. Then,

$$\begin{aligned} |\mathbf{x}(t) - \mathbf{x_n}(t)| &\leq |\mathbf{x}(t)| + |\mathbf{x_n}(t)| \\ |\mathbf{x}(t) - \mathbf{x_n}(t)| &\leq 2\gamma_x. \end{aligned} \tag{6.7}$$

Using these considerations on Eq. (6.6),

$$|\widetilde{\mathbf{w}}(t)| \leq \sigma_x(2\gamma_x) + |f(\mathbf{x}(t), \mathbf{u}(t-D)) - f(\mathbf{x_n}(t), \mathbf{u}(t-D))| + \overline{\gamma}.$$

Using Assumption 6.1, this last equation can be written as

$$|\widetilde{\mathbf{w}}(t)| \leq \sigma_x(2\gamma_x) + \sigma_x(|\mathbf{x}(t) - \mathbf{x_n}(t)|) + \overline{\gamma},$$

and, using Eq. (6.7) again,

$$|\widetilde{\mathbf{w}}| \leq 2\sigma_x(2\gamma_x) + \overline{\gamma}. \tag{6.8}$$

Also, consider that the function σ_x from Assumption 6.1 is defined as a constant, i.e., the function $f(.)$ is such that

$$|f(\mathbf{x}_1, \mathbf{u}) - f(\mathbf{x}_2, \mathbf{u})| \leq \lambda|\mathbf{x}_1 - \mathbf{x}_2|. \tag{6.9}$$

Although this simplification results in a more conservative bound, the analysis of Eq. (6.8) is simplified. Using this in Eq. (6.8) results in

$$|\widetilde{\mathbf{w}}| \leq 4\lambda\gamma_x + \overline{\gamma}.$$

Hence, it is not possible to affirm that the bound on $\widetilde{\mathbf{w}}$ is the same as the bound on $\overline{\mathbf{w}}$, thus, the robustness properties of the NLFSP using the system description given by Eq. (6.2) are not proven. This happens because the disturbance $\overline{\mathbf{w}}$ appears as a parameter of function $f(.)$ (see Appendix A.3), and it is necessary to take it out of $f(.)$ to make the proper interpretation, but, this is only possible using Assumption 6.1, which incorporates conservativeness in the results.

This problem motivated the use of a different nonlinear system description, as will be seen in Section 6.3.

6.3 AN ALTERNATIVE SYSTEM DESCRIPTION

As seen in Section 6.2.1, the robustness properties of the NLFSP for stable nonlinear systems could not be proven because of the system

description, which is given by Eq. (6.2). To avoid this problem, the following system description is utilized

$$\begin{cases} \begin{bmatrix} \mathbf{x_{n1}}(t+1) \\ \vdots \\ \mathbf{x_{np}}(t+1) \end{bmatrix} = \begin{bmatrix} f_1(\mathbf{x_{n1}}(t), \mathbf{u}(t-d_1)) \\ \vdots \\ f_p(\mathbf{x_{np}}(t), \mathbf{u}(t-d_p)) \end{bmatrix} \\ \begin{bmatrix} \mathbf{x_1}(t) \\ \vdots \\ \mathbf{x_p}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{x_{n1}}(t) \\ \vdots \\ \mathbf{x_{np}}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{w_1}(t) \\ \vdots \\ \mathbf{w_p}(t) \end{bmatrix} \end{cases}$$

or, in a simplified manner,

$$\begin{cases} \mathbf{x_n}(t+1) = f(\mathbf{x_n}(t), \mathbf{u}(t-D)) & \text{(6.10a)} \\ \mathbf{x}(t) = \mathbf{x_n}(t) + \mathbf{w}(t), & \text{(6.10b)} \end{cases}$$

where $\mathbf{x_n}$ now defines the nominal state-vector of the system, i.e., without disturbances, and $\mathbf{x_{ni}} \in \mathbb{R}^{n_i}$ is the ith group of states. The remaining variables are defined in the same way as in Section 6.1.

In this system description, the measured states $\mathbf{x}$ is affected by an output additive disturbance $\mathbf{w}$. The difference between the system descriptions given by Eq. (6.2) and Eq. (6.10) is that in the former the disturbance $\overline{\mathbf{w}}$ directly affects the evolution of the system, while in the latter $\mathbf{w}$ affects only the system output, without changing the nominal evolution of the system. These two descriptions are presented graphically in Figure 45. To better clarify the differences and similarities between these disturbances, they will be further analyzed in Sections 6.3.1 and 6.3.2.

6.3.1 State Disturbances

Consider that the nominal value of $\mathbf{x}(t+1)$ given the information at time t is[4]

$$\overline{\mathbf{x}}(t+1|t) = f(\mathbf{x}(t), \mathbf{u}(t-d_n)), \tag{6.11}$$

where d_n is the nominal dead time. However, the "real" system is given by

$$\mathbf{x}(t+1) = f_r(\mathbf{x}(t), \mathbf{u}(t-d)) + \mathbf{n}(t), \tag{6.12}$$

[4]For simplicity, only the case with one group of states will be considered.

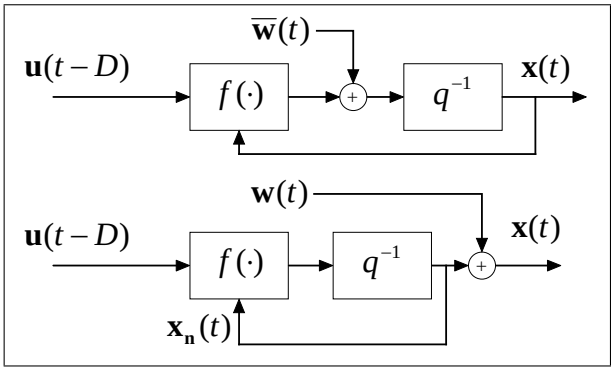

Figure 45 – Systems descriptions with the disturbance on the states and on the output, respectively.

where $\mathbf{n} \in \mathbb{R}^n$ is an additive external disturbance, and f_r is the function that describes the evolution of the system.

Then, the "real" output of the system can be described as

$$\begin{aligned}\mathbf{x}(t+1) =& \overline{\mathbf{x}}(t+1|t) + \overline{\mathbf{w}}(t)\\ \mathbf{x}(t+1) =& f(\mathbf{x}(t), \mathbf{u}(t-d_n)) + \overline{\mathbf{w}}(t).\end{aligned}$$

Then, substituting Eq. (6.12) in this last equation, the disturbance $\overline{\mathbf{w}}$ can be obtained

$$\overline{\mathbf{w}}(t) = f_r(\mathbf{x}(t), \mathbf{u}(t-d)) - f(\mathbf{x}(t), \mathbf{u}(t-d_n)) + \mathbf{n}(t).$$

That is, the disturbance $\overline{\mathbf{w}}(t)$ represents the difference between the real value of the states of the system at time $t+1$ and the expected value at time $t+1$ given the information at time t. Also note that the first two terms of the right side of this last equation represent the model uncertainties, and the last one represents the external disturbance.

6.3.2 Output Disturbances

Before the output disturbance can be discussed, some notations must be introduced to facilitate the interpretation of the results. First, given a signal $\mathbf{a} \in \mathbb{R}^{n_a}$, the signal sequence is denoted by

$$\mathbf{a}_{[i,j]} \triangleq \{\mathbf{a}(i), \mathbf{a}(i+1), \ldots, \mathbf{a}(j)\},$$

and, with a slight abuse of notation, sometimes $\mathbf{a}$ will also denote a sequence, where the cardinality of the sequence is inferred from the context. Also, $\mathbf{0}_{[i,j]}$ denotes a suitable signal sequence taking a null value.

The solution of the following system

$$\mathbf{x}(t+1) = f(\mathbf{x}(t), \mathbf{u}(t-d)) + \overline{\mathbf{w}}(t), \tag{6.13}$$

given an initial state $\mathbf{x}(t)$, a sequence of inputs $\mathbf{u}$ and disturbances $\overline{\mathbf{w}}$ at sampling time $t+j$, is denoted by

$$\mathbf{x}(t+j) = \phi(j, \mathbf{x}(t), \mathbf{u}_{[t-d,t-d+j-1]}, \overline{\mathbf{w}}_{[t,t+j-1]}). \tag{6.14}$$

Note that this solution can be obtained using Eq. (6.13) recursively from time $t+1$ until time $t+j$.

For example,

$$\begin{aligned}\mathbf{x}(t+2) =& f(f(\mathbf{x}(t), \mathbf{u}(t-d)) + \overline{\mathbf{w}}(t), \mathbf{u}(t-d+1)) + \overline{\mathbf{w}}(t+1)\\ =& \phi(2, \mathbf{x}(t), \mathbf{u}_{[t-d,t-d+1]}, \overline{\mathbf{w}}_{[t,t+1]})\end{aligned}$$

Now, the system description with output disturbance is given by

$$\begin{cases} \mathbf{x_n}(t+1) = f(\mathbf{x_n}(t), \mathbf{u}(t-d_n)) & \text{(6.15a)}\\ \mathbf{x}(t) = \mathbf{x_n}(t) + \mathbf{w}(t), & \text{(6.15b)}\end{cases}$$

where $\mathbf{x_n} \in \mathbb{R}^n$ describes the nominal evolution of the system, and $\mathbf{w} \in \mathbb{R}^n$ is the additive output disturbance which can describe model uncertainties and external perturbations.

From Eq. (6.15), the disturbance $\mathbf{w}$ is given by

$$\mathbf{w}(t) = \mathbf{x}(t) - \mathbf{x_n}(t),$$

and, considering that the "real" evolution of the system follows Eq. (6.12),

$$\mathbf{w}(t) = f_r(\mathbf{x}(t-1), \mathbf{u}(t-d-1)) + \mathbf{n}(t-1) - f(\mathbf{x_n}(t-1), \mathbf{u}(t-d_n-1)).$$

Defining

$$\phi_r(t, \mathbf{x}(0), \mathbf{u}, \mathbf{n}),$$

as the evolution of the system given by Eq. (6.12), and

$$\phi(t, \mathbf{x_n}(0), \mathbf{u}, \mathbf{0}), \tag{6.16}$$

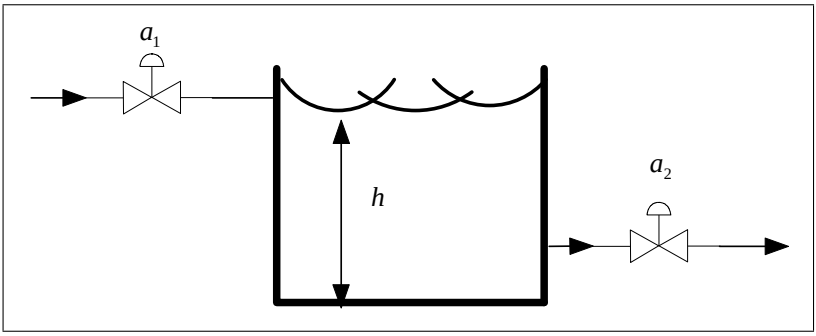

Figure 46 – Schematic drawing of a tank.

as the evolution of the nominal system (Eq. (6.15a)), then

$$\mathbf{w}(t) = \phi_r(t, \mathbf{x}(0), \mathbf{u}_{[-d,t-d-1]}, \mathbf{n}_{[0,t-1]}) - \phi(t, \mathbf{x_n}(0), \mathbf{u}_{[-d_n,t-d_n-1]}, \mathbf{0}),$$

where it is considered that $\mathbf{x_n}(0) = \mathbf{x}(0)$. Hence, $\mathbf{w}(t)$ represents the difference between the nominal evolution of the system at time t and the "real" evolution of the system at the same time instant. Naturally, $\mathbf{w}$ can assume higher values than $\overline{\mathbf{w}}$, because, while $\overline{\mathbf{w}}$ is a one time sample difference between the expected and "real" values of the system output, $\mathbf{w}$ represents the difference between the evolution of the output of the nominal and "real" systems in a given interval. However, they are simply different interpretation of a phenomenon, i.e., the effect of the system uncertainties and external disturbances on the output.

Nonetheless, the concept of ISS can still be applied to the system with output disturbances (see Appendix C.2). Hence, the disturbance $\mathbf{w}$ has a role analogous $\overline{\mathbf{w}}$, i.e., if its bound is large, it will be harder to guarantee closed-loop stability.

6.3.3 Example

To make the role of the disturbances $\overline{\mathbf{w}}$ and $\mathbf{w}$ more clear, consider the control of the level of a cylindrical tank, shown in Figure 46. The behaviour of the tank level is described by the following differential equation

$$A\frac{dh(t)}{dt} = k_1 a_1(t) - k_2 a_2(t)\sqrt{2gh(t)},$$

where h is the level inside the tank, A is the base area of the tank, g is the gravitational acceleration, k_1 and k_2 are constants which relate the opening of the respective valve a_i with the input and output flows, respectively. The manipulated variable is a_2, and the external disturbance is represented by a_1. The discretized nonlinear model of this system can be obtained using the forward approximation of the derivative with sampling period T_s, which results in

$$h(t+1) = h(t) + \frac{T_s}{A}\left(k_1 a_1(t) - k_2 a_2(t)\sqrt{2gh(t)}\right).$$

However, the value of the disturbance a_1 is not known, but it is estimated as a fixed value $\overline{a}_1$, hence, $a_1(t) = \overline{a}_1 + \delta_a(t)$, and also, the value of k_2 is uncertain, i.e., $k_2 = \overline{k}_2 + \delta_k$. The process equation is then

$$\begin{aligned} h(t+1) =& h(t) + \frac{T_s}{A}\left(k_1(\overline{a}_1 + \delta_a(t)) - (\overline{k}_2 + \delta_k)a_2(t)\sqrt{2gh(t)}\right) \\ h(t+1) =& h(t) + \frac{T_s}{A}\Big(k_1\overline{a}_1 - \overline{k}_2 a_2(t)\sqrt{2gh(t)} \\ & + k_1\delta_a(t) - \delta_k a_2(t)\sqrt{2gh(t)}\Big) \end{aligned}$$

or, in a simplified manner, using the system description with state disturbances,

$$h(t+1) = f(h(t), a_2(t)) + \overline{w}(t), \tag{6.17}$$

where

$$f(h(t), a_2(t)) = h(t) + \frac{T_s}{A}\left(k_1\overline{a}_1 - \overline{k}_2 a_2(t)\sqrt{2gh(t)}\right),$$

is the nominal system equation, and

$$\overline{w}(t) = \frac{T_s}{A}\left(k_1\delta_a(t) - \delta_k a_2(t)\sqrt{2gh(t)}\right),$$

is the system state disturbance, where the first term of the right-hand side of the equation represents the external disturbance and the second term the model uncertainties.

The same process can also be described using the alternative system description

$$\begin{cases} h_n(t+1) = f(h_n(t), a_2(t)) & \text{(6.18a)} \\ h(t) = h_n(t) + w(t), & \text{(6.18b)} \end{cases}$$

where h_n represents the nominal evolution of the system. Considering that the nominal evolution of the system can also be represented as (using the notation introduced in Eq. (6.16))

$$h_n(t) = \phi(t, h_n(0), \mathbf{a_2}_{[0,t-1]}, \mathbf{0}),$$

and that, given Eq. (6.17), the "real" system output can also be described as

$$h(t) = \phi(t, h(0), \mathbf{a_2}_{[0,t-1]}, \overline{\mathbf{w}}_{[0,t-1]}),$$

then, from Eq. (6.18b), the output disturbance is given by

$$w(t) = \phi(t, h(0), \mathbf{a_2}_{[0,t-1]}, \overline{\mathbf{w}}_{[0,t-1]}) - \phi(t, h_n(0), \mathbf{a_2}_{[0,t-1]}, \mathbf{0}),$$

where it is considered that $h(0) = h_n(0)$.

An open-loop simulation of the process was made where the input and the disturbance are varied. The system parameters are: $A = 0.5$ [m^2], $k_1 = 0.1$ [m^3/s], $k_2 = 0.05$ [m^3/s], $\delta_k = 0.005$ [m^3/s], $g = 9.81$ [m/s^2], $T_s = 1$ [s], $\overline{a}_1 = 0.5$, $a_2(0) = 0.3$. The output, input and disturbance values during the simulation are shown in Figure 47. There are three curves in the output plot in this figure, the "real" output, the nominal output, and the predicted output. The first is the actual value of the system output. The second is the evolution considering the nominal system (Eq. (6.18a)), and the last one is the open-loop prediction of the system output, i.e., the first term of the right-hand side of Eq. (6.17). During the simulation, initially the system exhibits some dynamics because of the initial conditions, then, at 300 [s], the input a_2 is changed from 0.3 to 0.4. Then, at 600 [s], the disturbance a_2 is changed from 0.5 to 0.65 ($\delta_a = 0.15$).

These different system responses were computed to give a better understanding of the state and output disturbances, $w(t)$ and $\overline{w}(t)$, respectively, which are shown in Figure 48. Before the change in a_1, the disturbances represent only model uncertainties. Note how the nominal response diverges from the "real" output. The difference between the two is the disturbance w. Also note that the values of $\overline{w}$, which is the difference between the predicted output and the "real" one, are considerably smaller than those of w. This happens because the predicted output is computed using the past "real" values of the output, hence the state disturbance is smaller than the output disturbance. This difference is more prominent after the disturbance change, because the predicted output uses the effect of the disturbance on the past "real" output values to make a prediction, while, in the nominal case, no in-

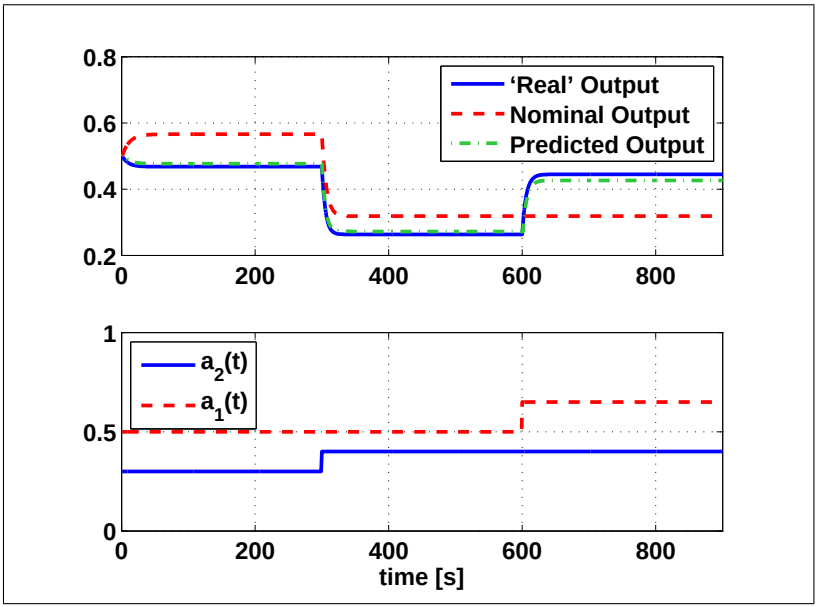

Figure 47 – Open-loop output of the tank process.

formation of the disturbance is known, hence the nominal output does not change, which implies bigger values of the disturbance w. However, they are simply different interpretations of a phenomenon, i.e., the effect of the model uncertainties and external disturbances on the output.

6.3.4 Dead-time free model using the alternative system description

Before the optimal predictor and the NLFSP can be analyzed using the alternative system description, it is first necessary to define the dead-time-free auxiliary system, as was done previously:

$$\begin{cases} \overline{\mathbf{x}}(t+1) = f(\overline{\mathbf{x}}(t), \mathbf{u}(t)) & \text{(6.19a)} \\ \widetilde{\mathbf{x}}(t) = \overline{\mathbf{x}}(t) + \widetilde{\mathbf{w}}(t), & \text{(6.19b)} \end{cases}$$

where $\overline{\mathbf{x}}(t) \triangleq \mathbf{x_n}(t+D)$, $\widetilde{\mathbf{w}} \in \mathbb{R}^n$ is the output equivalent disturbance, and $\widetilde{\mathbf{x}}(t) \triangleq \mathbf{x}(t+D|t)$ is the output of the predictor structure.

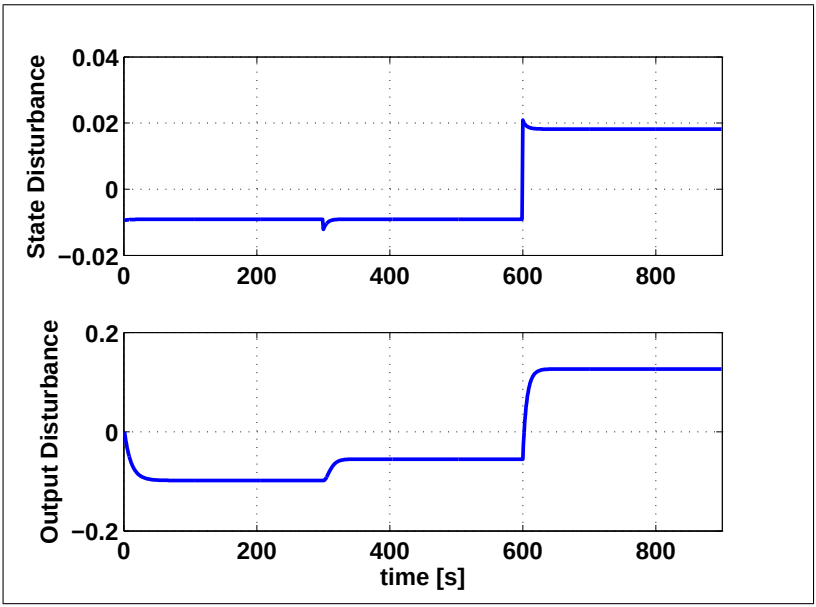

Figure 48 – Values of the state ($\overline{w}$) and output (w) disturbances.

Hence, the primary controller is tuned considering the dead-time-free system, thus, by the ISS concept (Appendix C.2), the bound on $\widetilde{\mathbf{w}}$ plays an important role considering the closed-loop robustness. Moreover, as in Chapter 5, since the interest lies in the predictor structure, the following assumption must hold for the primary controller:

Assumption 6.3. *The primary controller* $\kappa(\widetilde{\mathbf{x}}(t), \mathbf{r}(t))$ *was designed for the dead-time-free auxiliary model, which is given by Eq. (6.19), in a way that, by the local-ISS concept (see Appendix C), the closed-loop dead-time-free system remains stable if the states, inputs and disturbances are bounded, i.e.,* $\widetilde{\mathbf{x}} \in \mathbb{X} = \{\widetilde{\mathbf{x}} \in \mathbb{R}^n : |\widetilde{\mathbf{x}}| \leq \lambda_x\}$, $\mathbf{u} \in \mathbb{U} = \{\mathbf{u} \in \mathbb{R}^{m \times l} : |\mathbf{u}| \leq \lambda_u\}$ *and* $\widetilde{\mathbf{w}} \in \widetilde{\mathbb{W}} = \{\widetilde{\mathbf{w}} \in \mathbb{R}^n : |\widetilde{\mathbf{w}}| \leq \tilde{\gamma}\}$, *where the bounds* λ_x, λ_u *and* $\tilde{\gamma}$ *are not necessarily known but are greater than zero.*

6.4 OPTIMAL PREDICTOR ANALYSIS

Despite the representation of uncertainties and perturbations as output disturbances, optimal predictors still estimate the disturbance

as an integrated white noise in the states:

$$\mathbf{x}(t+1) = f(\mathbf{x}(t), \mathbf{u}(t)) + \frac{\mathbf{e}(t)}{\Delta} \tag{6.20}$$

where $\mathbf{e}(t)$ is a white noise with zero mean, and $\Delta = 1 - q^{-1}$ is included to allow the representation of constant disturbances [2].

For the analysis of the effect of the optimal predictor on the closed-loop robustness, the following theorem is needed.

Theorem 6.2. *Consider an open-loop stable nonlinear dead-time system (Assumption 6.2) described by Eq. (6.10) and the dead-time-free model given by Eq. (6.19), where the function $f(.)$ follows Assumption 6.1, and there is only one group of states*[5]*, i.e., $p = 1$. If the optimal predictor, which utilizes the model described by* (6.20)*, is used, the equivalent disturbance at time t is bounded by*

$$|\widetilde{\mathbf{w}}(t)| \leq c_d(|\mathbf{w}(t)|),$$

where c_j is a recursive $\mathcal{K}$-function given by

$$c_j(|\mathbf{w}(t)|) = |\mathbf{w}(t)| + \sigma_x(|\mathbf{w}(t-1)) + \sigma_x(c_{j-1}(|\mathbf{w}(t)|)), \tag{6.21}$$

with $c_0(|\mathbf{w}(t)|) = |\mathbf{w}(t)|$.

Proof. See Appendix A.4. □

By Theorem 6.2, the bound of the equivalent disturbance of the optimal predictor is dependent on the nominal dead time of the system. Suppose again that $\mathbf{w}$ is bounded, i.e., $|\mathbf{w}| \leq \gamma$, and that the function σ_x from Assumption 6.1 is defined as a constant, as was done in Eq. (6.9), to simplify the analysis. Then, from (6.21), and considering the bound γ on $\mathbf{w}$, for $j = 1$,

$$c_1(\gamma) = \gamma + \lambda\gamma + \lambda\gamma = \gamma + 2\lambda\gamma$$

For $j = 2$,

$$c_2(\gamma) = \gamma + \lambda\gamma + \lambda c_1(\gamma) = \gamma + 2\lambda\gamma + 2\lambda^2\gamma.$$

[5]The requirement of only one group of states is done to simplify the analysis of the optimal predictor. The results can be extended to the more general case, but the interpretation would be more difficult, although the result would be the same, i.e., that the bound on the equivalent disturbance $\widetilde{\mathbf{w}}$ is dependent on the nominal dead time for optimal predictors.

By inspection,

$$c_d(\gamma) = \gamma + 2\lambda\gamma + 2\lambda^2\gamma + \ldots + 2\lambda^d\gamma = \widetilde{\gamma}.$$

Then, by virtue of Theorem 6.2, if $\mathbf{w}$ is bounded, the equivalent disturbance is also bounded, i.e., $\widetilde{\mathbf{w}} \in \widetilde{\mathbb{W}} = \{\widetilde{\mathbf{w}} \in \mathbb{R}^n : |\widetilde{\mathbf{w}}| \leq \widetilde{\gamma}\}$. It is possible to verify if $\widetilde{\gamma}$ can be smaller than γ analyzing the following inequality

$$\begin{aligned} \widetilde{\gamma} = \gamma + 2\lambda\gamma + 2\lambda^2\gamma + \ldots + 2\lambda^d\gamma \leq & \gamma \\ \Rightarrow \quad 2\gamma(\lambda + \lambda^2 + \ldots + \lambda^d) \leq & 0. \end{aligned}$$

This last inequality would only hold if $\gamma \leq 0$ or $\lambda \leq 0$, but, by definition they are positive definite, hence, $\widetilde{\gamma} \geq \gamma$, i.e., the bound on $\widetilde{\mathbf{w}}$ will always be bigger than the bound on $\mathbf{w}$ (except when $d = 0$, then the bounds are equal), and dependent on the nominal dead time.

Then, again, the robust behaviour of the system will change with the nominal dead time if the optimal predictor is used, as was expected, since this is an intrinsic property of this type of predictor scheme, thus this aspect of the predictor should not change just because the system description changed. Hence, all the considerations regarding the optimal predictor done in Section 5.3 remain true: (i) in the design of the primary controller the nominal dead time must be taken into account, (ii) robustness will be compromised, (iii) noise effects will be amplified. The addition of the filter polynomial $\mathbf{T}(q)$ to help with these problems is also possible, but the dependency on the nominal dead time still remains and the tuning of $\mathbf{T}(q)$ is not trivial.

6.5 ANALYSIS OF THE NLFSP USING THE ALTERNATIVE SYSTEM DESCRIPTION

The NLFSP scheme is still given by Eq. (6.5), and its block diagram is shown in Figure 44. For the linear case, it was shown that if the predictor filter has low-pass characteristics, the robustness of the system is increased [1, 3]. This property also holds for the proposed NLFSP. To prove this statement, the following theorem is used.

Theorem 6.3. *Consider an open-loop stable nonlinear dead-time system (Assumption 6.2) described by Eq. (6.10) and the dead-time free model given by Eq. (6.19). If the NLFSP structure is used, which is*

described by Eq. (6.5), the equivalent disturbance is

$$\widetilde{\mathbf{w}}(t) = \mathbf{F_r}(q)\mathbf{w}(t), \tag{6.22}$$

with $\mathbf{F_r}(q) = F_r(q)\mathbf{I}$*, where* $F_r(q)$ *is a SISO filter.*

Proof. See Appendix A.5. □

Note that this is exactly the same result presented in Section 5.4.1 for the NLFSP for systems with input nonlinearities, with the exception that now $\mathbf{w}$ represents an output disturbance. Hence, all the considerations done in Section 5.4.1 are still valid.

Given that $\mathbf{w}$ is bounded, i.e., $\mathbf{w} \in \mathbb{W} = \{\mathbf{w} \in \mathbb{R}^n : |\mathbf{w}| \leq \gamma\}$, the bound on $\widetilde{\mathbf{w}}$ will also be γ as long as the filter has unitary static gain $F_r(1) = 1$ and has monotonic behaviour, i.e., it has real poles and does not have dominant zeros, or, if no filter is used $\mathbf{F_r}(q) = \mathbf{I}$.

Therefore, the NLFSP, with an appropriate filter tuning, maintains the bound on $\mathbf{w}$, which can be used to improve robustness, if compared to the optimal predictor, which increased the original bound. Furthermore, since $\mathbf{w}(t)$ can represent model mismatch dynamics and noise, and that these type of signals are usually in the range of medium and high frequencies, when $F_r(q)$ is tuned as a low-pass filter their effects will be attenuated with the use of the filter, improving the overall robustness. The tuning procedure to improve robustness is exactly the one described in Section 5.4.5, which uses Fourier analysis to estimate the frequency characteristics of $\mathbf{w}$ from simulation or experimental data.

Regarding the internal stability of the predictor, by Assumption 6.2, the system only has stable equilibrium points in the domain of interest $\mathbb{X} \times \mathbb{U}$. Hence, as long as the predictor states $\widetilde{\mathbf{x}}$ and $\mathbf{x_n}$ are contained in the domain $\mathbb{X}$, and $\mathbf{u} \in \mathbb{U}$, the predictor is stable. Since the controller $\kappa(\widetilde{\mathbf{x}}(t), \mathbf{r}(t))$, by Assumption 6.3, does exactly this as long as $|\widetilde{\mathbf{w}}| \leq \tilde{\gamma}$, the predictor is stable.

The next section will present an example to illustrate the advantages of the NLFSP and corroborate the theoretical results.

6.6 AN ILLUSTRATIVE EXAMPLE

Consider the following discrete nonlinear system introduced in [108]

$$\begin{cases} \mathbf{x}(t+1) = & f(\mathbf{x}(t), u(t-d)) \\ y(t) = & [1\ 0]\,\mathbf{x}(t) + w(t) \end{cases}, \tag{6.23}$$

where $y, u, w \in \mathbb{R}$ are the output, the input and the disturbance of the system, respectively, $\mathbf{x}(t) = [y(t), y(t-1)]^T$, $d = 7$ is the nominal dead time, and $f(.) : \mathbb{R}^2 \times \mathbb{R} \to \mathbb{R}^2$ is defined as

$$f(\mathbf{x}(t), u(t-d)) = \begin{bmatrix} \frac{2.5y(t)y(t-1)}{1+y(t)^2+y(t-1)^2} + 1.2u(t-d) \\ +0.3\cos(0.5(y(t)+y(t-1)) \\ y(t) \end{bmatrix} \tag{6.24}$$

The control of this process was made using PNMPC [83], which were described in Section 3.3.4. The simulation results will consider the original PNMPC, which uses the optimal predictor described earlier, and the PNMPC with the proposed NLFSP. The control parameters are: $N_1 = d + 1$, $N_2 = d + 5$, $N_u = 3$ and $Q_u = 5$. These parameters will give a settling time of approximately 6 time samples. The nominal response of the PNMPC with the optimal predictor and with the NLFSP ($\mathbf{F_r}(q) = \mathbf{I}$) are shown in Figure 49. There is a set-point change from 0 to 1 at sample time 20, and at time 60 an input disturbance of 0.1 is applied. Also, a zero-mean white noise with standard deviation of 0.03 was added after time 100 to see the effect of the noise on the closed-loop system. Notice that the response with both controllers are very similar in the nominal case.

Now consider the case where the real value of the dead time is $d_r = 8$, i.e., a one time sample mismatch. The simulation results are shown in Figure 50. Notice that in the case of the optimal predictor, there is an oscillatory behaviour because of the dead-time mismatch, which is much less prominent with the NLFSP. This happens, as discussed earlier, because of how the predictor uses the disturbance information to compute the prediction after the dead time. The reason for this becomes clear with the analysis of the disturbances, which are shown in Figure 51.

Analysing the simulation data of the optimal predictor case, $|\mathbf{w}| \leq \gamma = 0.54$ and $|\widetilde{\mathbf{w}}| \leq 1.15$, i.e., the bound of the equivalent disturbance is amplified with the optimal predictor. This has a clear impact on the closed-loop robustness of the system, as indicated by the

presence of oscillatory behaviour on the system's response. This amplification of the disturbance does not happen with the NLFSP, as can be observed from the second plot of Figure 51. Also note that $\mathbf{w} = \widetilde{\mathbf{w}}$ and $|\mathbf{w}| < 0.42$, i.e., the disturbance is not amplified, which contributes to the overall robustness, resulting in a less evident oscillatory behaviour of the system's response.

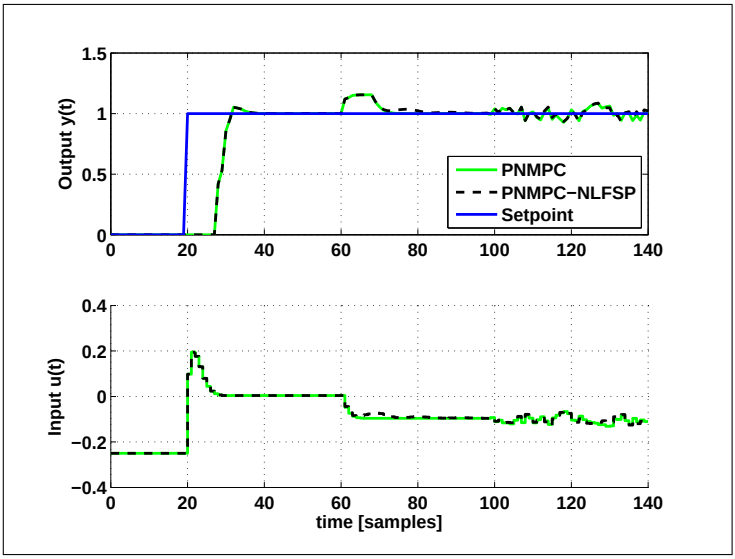

Figure 49 – Nominal response of the system with different predictors.

The robustness and response of the system can still be improved if a low-pass filter is used with the NLFSP. From the simulation data of the PNMPC-NLFSP case the frequency characteristics of $\mathbf{w}$ is found through a Fourier analysis of the signal, which is presented in Figure 52. Notice that there are components with considerable magnitude around and after 1 [rad/s]. To attenuate their effect, the following filter was designed

$$\mathbf{F_{r2}}(q) = \frac{0.2q^{-1}}{1 - 0.8q^{-1}}\mathbf{I}.$$

In Figure 50 the response of the system with this filter is shown (identified as PNMPC-NLFSP2). In this case the system's response is much smoother. Analysing the disturbance data, the values of the bounds are obtained: $|\mathbf{w}| \leq 0.42$, $|\widetilde{\mathbf{w}}| \leq 0.25$, i.e., the bound on $\widetilde{\mathbf{w}}$ is smaller than the bound on $\mathbf{w}$. This happens because the bound

on $\mathbf{w}$ must account for the peak values of $\mathbf{w}$, but, as $\widetilde{\mathbf{w}}$ is $\mathbf{w}$ filtered by a low-pass filter, the peaks are cut off, reducing the bound on $\widetilde{\mathbf{w}}$. These peaks come mostly from the model mismatch (simulation results before time sample 100), and noise effects (simulation results after time sample 100).

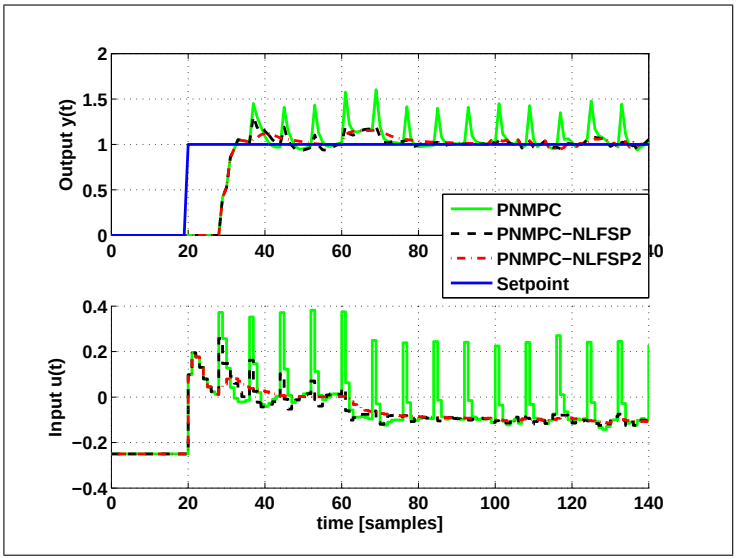

Figure 50 – Response of the system with dead-time mismatch with different predictors.

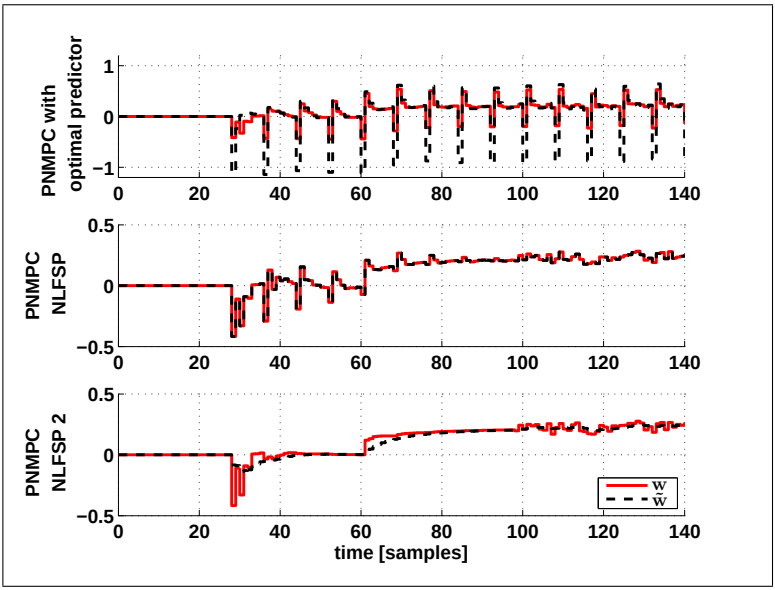

Figure 51 – Values of the disturbances in the case with dead-time mismatch with different predictors.

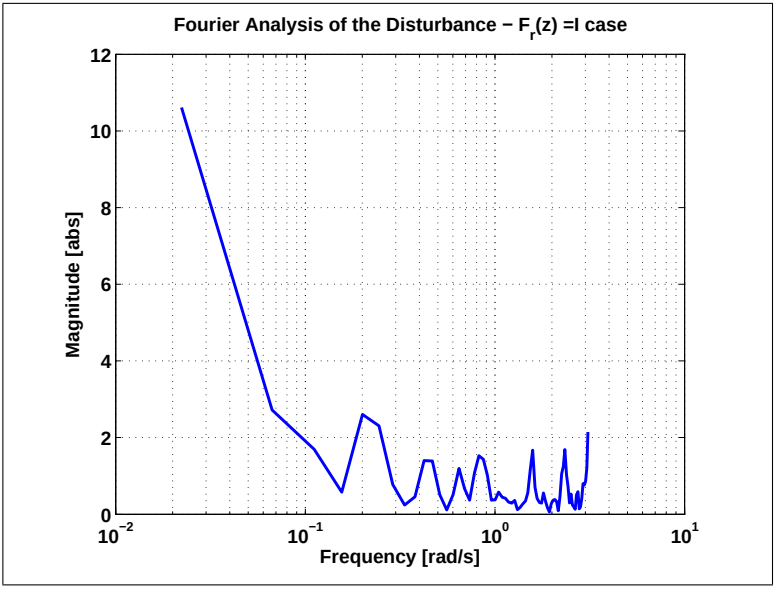

Figure 52 – Frequency characteristic of the disturbance $\mathbf{w}$ obtained through Fourier analysis for the case where $\mathbf{F_r}(z) = \mathbf{I}$.

6.7 SIMULATED CASE STUDY: A MIMO CSTR

In this section, a MIMO CSTR will be presented. Differently from the CSTR used in Section 5.6, in this case the temperature varies, affecting the chemical reactions inside the CSTR. The MIMO CSTR case will be based on the one introduced in [109], where the output variables are the temperature inside the reactor (T), and the concentration of the product A (C_a). The input variables are the process flow rate (F), and the coolant flow rate (F_c). The unmeasurable disturbances are the temperature of the feed (T_f), the coolant temperature (T_{cf}) and the concentration of product A in the feed flow (C_{af}). The differential equations describing the process dynamics are:

$$\begin{cases} \dfrac{dC_a(t)}{dt} = \dfrac{F(t-d_{c1})}{V}[C_{af}(t) - C_a(t)] - k_0 C_a(t) e^{-E/RT(t)} & \text{(6.25a)} \\ \dfrac{dT(t)}{dt} = \dfrac{F(t-d_{c1})}{V}[T_f(t) - T(t)] + k_1 C_a(t) e^{-E/RT(t)} \\ \qquad + k_2 F_c(t-d_{c2})[1 - e^{-k_3/F_c(t-d_{c2})}][T_{cf}(t) - T(t)] & \text{(6.25b)} \end{cases}$$

where the parameters are given in Table 1, along with the values of the inputs and outputs at the chosen operating point as defined in [109]. The system also exhibits input delays $d_{c1} = 1$ [min] and $d_{c2} = 1.4$ [min]. The constants k_i are given below, along with its dimensions

$$k_1 = -\frac{\Delta H k_0}{\rho C_p} = 1.44 \times 10^{13} \text{ [l K/mol min]},$$

$$k_2 = \frac{\rho_c C_{pc}}{\rho C_p V} = 0.01 \text{ [l}^{-1}\text{]},$$

$$k_3 = \frac{h_a}{\rho_c C_{pc}} = 700 \text{ [l/min]}.$$

To control the process, the PNMPC algorithm will be used. The model used by the controller is obtained discretizing the system equations (Eq. (6.25)) using the forward approximation of the derivative, with a sampling time of $T_s = 0.1$ [min][6]:

[6]For this system, the forward approximation resulted in a good model with this sampling time. However, this method of approximation, which generally results in a simpler nonlinear model, needs a relatively small sample time to work adequately [105].

Table 1 – Parameters and steady-state values of the MIMO CSTR

Measured product concentration	C_a	0.1 [mol/l]
Coolant flow rate	F_c	103.41 [l/min]
Feed concentration	C_{af}	1 [mol/l]
Inlet coolant temperature	T_{cf}	350 [K]
Heat transfer term	h_a	7×10^5 [cal/min K]
Activation energy term	E/R	1×10^4 [K]
Liquid densities	ρ, ρ_c	1×10^3 [g/l]
Reactor temperature	T	438.51 [K]
Process flow rate	F	100 [l/min]
Feed temperature	T_f	350 [K]
CSTR Volume	V	100 [l]
Reaction rate constant	k_0	7.2×10^{10} [min^{-1}]
Heat of reaction	ΔH	-2×10^5 [cal/mol]
Specific Heats	C_p, C_{pc}	1 [cal/g K]

$$\begin{cases} C_a(t+1) = C_a(t) + T_s\left(\dfrac{F(t-d_1)}{V}[C_{af}(t) - C_a(t)]\right) \\ \qquad\qquad - T_s\left(k_0 C_a(t) e^{-E/RT(t)}\right) \\ T(t+1) = T(t) + T_s\left(\dfrac{F(t-d_1)}{V}[T_f(t) - T(t)]\right) \\ \qquad\qquad + T_s\left(k_1 C_a(t) e^{-E/RT(t)}\right) \\ \qquad\qquad + T_s\left(k_2 F_c(t-d_2)[1 - e^{-k_3/F_c(t-d_2)}][T_{cf}(t) - T(t)]\right) \end{cases}$$

The discrete dead times are $d_1 = 10$ and $d_2 = 14$ samples. Also, since the disturbances are unmeasurable, the nominal model used by the PNMPC controller will consider that the disturbances are fixed at the steady-state values given in Table 1. This system can be rewritten using the alternative system description:

$$\begin{cases} \mathbf{x_n}(t+1) = f(\mathbf{x_n}(t), \mathbf{u}(t-d)) \\ \qquad \mathbf{x}(t) = \mathbf{x_n}(t) + \mathbf{w}(t), \end{cases}$$

where $\mathbf{x}(t) = [C_a(t), T(t)]^T$, $\mathbf{u}(t) = [F(t), F_c(t)]^T$, $\mathbf{w}(t) = [w_1(t) w_2(t)]$, and $d = 10$ is the minimal dead time. Note that in this case it is

not possible to separate the system in groups of states because all the system states depend on each other. However, as will be demonstrated in the simulations, the obtained results are in accordance with the theoretical results obtained in this chapter.

Again, the PNMPC controller will be used with the optimal and the NLFSP predictors. The tuning parameters were chosen in a way that the closed-loop system reaches the steady-state values in less than 10 minutes for both outputs. To achieve the specifications, the tuning parameters values are: initial prediction horizon $N_1 = 10$ (to avoid the minimal dead time), $N_2 = 110$ as the final horizon, control horizon of $N_u = 30$ samples, and the weights of the error and of the control increments are, respectively, $Q_y = [1000, 1]$ and $Q_u = [3, 3]$. For the PNMPC-NLFSP, only the prediction horizons change, since the PNMPC will control the equivalent dead-time-free model, hence, $N_1 = 1$ and $N_2 = 100$.

All the simulation follow the same guidelines. Initially the system is at steady-state with $C_a = 0.1$ [mol/l] and $T = 438.51$ [K], then the set-point of C_a is changed to 0.05 [mol/l] at 2 [min], and the temperature T set-point is changed to 443.51 [K]. Then, at 25 [min], both set-points return to the original operating point. At 40 [min], the value of the inlet coolant temperature is raised from 350 to 355 [K] to simulate an external perturbation. Also, the MIMO CSTR dynamics were simulated using its differential equations given by Eq. (6.25). The nominal responses of the closed-loop system with both controllers are shown in Figure 53, where it can be seen that the original specifications are satisfied. The values of the disturbances $\mathbf{w}$ and the equivalent disturbances $\widetilde{\mathbf{w}}$ are shown in Figure 54. The plots were split in different time intervals because the values of the disturbances change dramatically when the external perturbation is applied on the system. One important fact, although in this case no uncertainties were added to the model, there is a discretization error because the model is simulated with a smaller sampling time. Hence, the disturbance $\mathbf{w}$ is not null, although it has small values before the external perturbation is applied.

After the analysis of the disturbance data, the theoretical results are again confirmed. For the optimal predictor, the bound on $\widetilde{\mathbf{w}}$ is bigger than the bound on $\mathbf{w}$: before the external perturbation ($t < 40$ [min]), $|\mathbf{w}| \leq 0.4665$ and $|\widetilde{\mathbf{w}}| < 0.5074$, and after ($t \geq 40$ [min]), $\mathbf{w}| \leq 6.3602$ and $|\widetilde{\mathbf{w}}| < 6.6107$. For the NLFSP case, since $\widetilde{\mathbf{w}} = \mathbf{w}$, their bounds are equal and have the values 0.4615 and 6.7338, before and after the external perturbation, respectively.

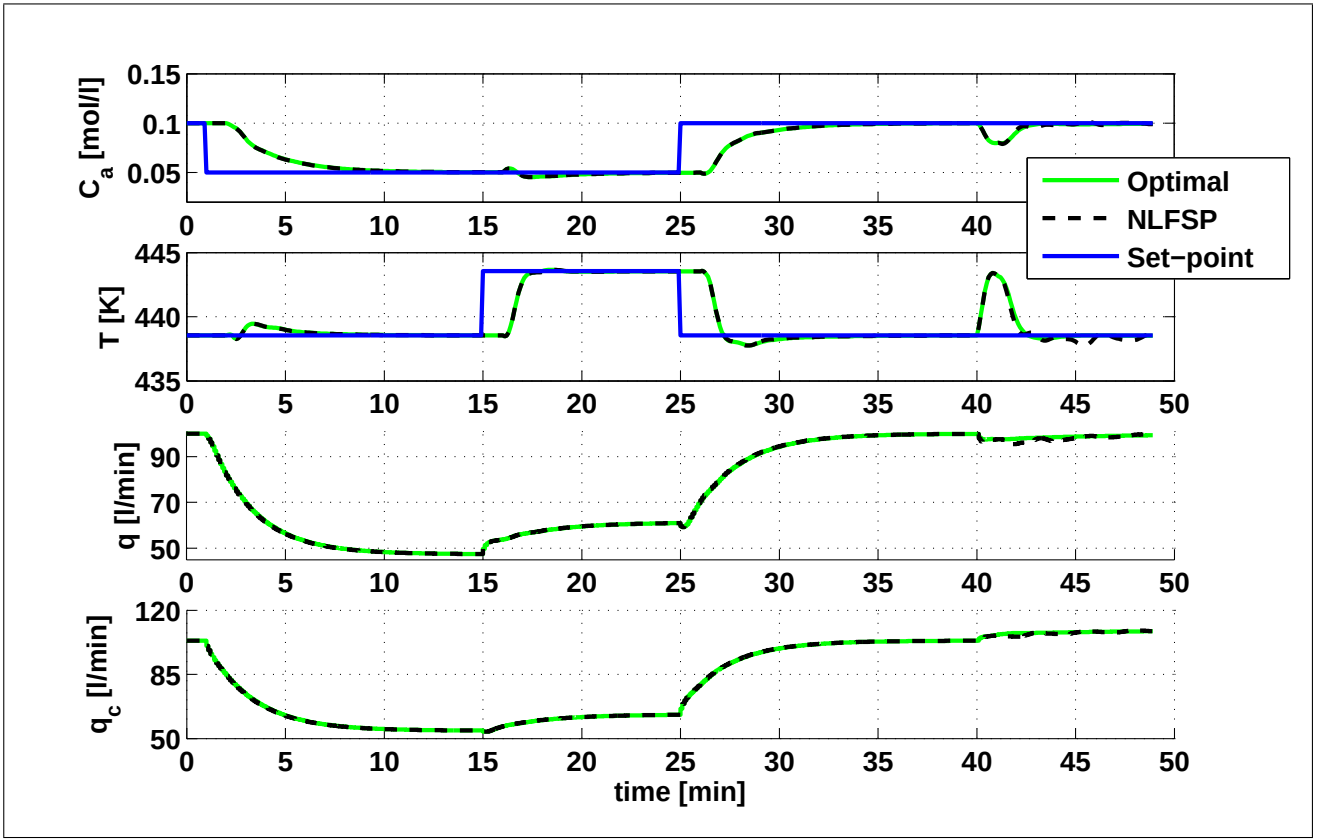

Figure 53 – Nominal closed-loop response of the MIMO CSTR.

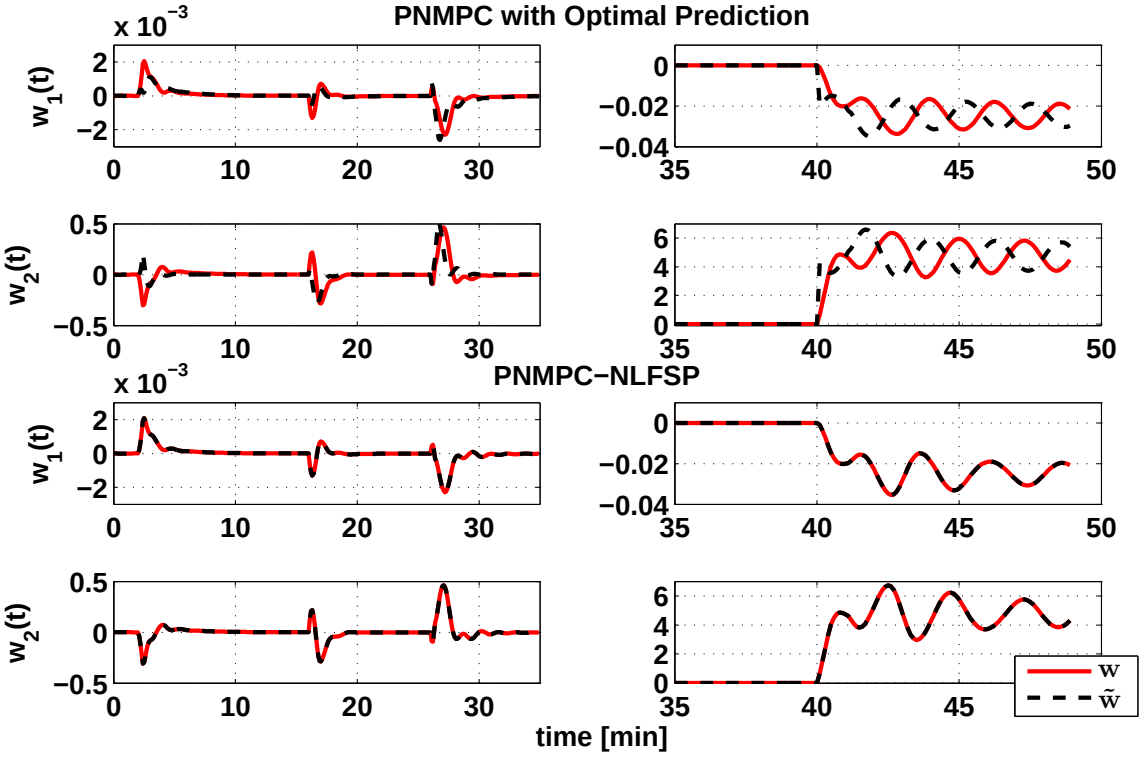

Figure 54 – Disturbances of the nominal closed-loop MIMO CSTR.

Table 2 – Bounds on the disturbances $\mathbf{w}$ and $\widetilde{\mathbf{w}}$ during the MIMO CSTR simulation scenarios with dead-time mismatch.

		$\lvert\mathbf{w}\rvert$	$\lvert\widetilde{\mathbf{w}}\rvert$	$\lvert\widetilde{\mathbf{w}}\rvert/\lvert\mathbf{w}\rvert$
Optimal	$t < 40$ [min]	1.6953	7.1857	4.24
	$t \geq 40$ [min]	6.4259	7.1857	1.12
NLFSP	$t < 40$ [min]	1.4693	1.4693	1
	$t \geq 40$ [min]	6.6691	6.6691	1
NLFSP with $\mathbf{F_{r1}}(z)$	$t < 40$ [min]	1.4890	0.6031	0.40
	$t \geq 40$ [min]	4.8861	4.7156	0.96

Now a dead-time mismatch is introduced to verify the behaviour of the closed-loop system in the presence of model uncertainties. The "real" delays are: $d_1 = 8$ and $d_2 = 12$ samples, i.e., the nominal dead times were overestimated in 2 samples for both inputs. The closed-loop responses are shown in Figure 55, and the disturbances in Figure 56. Notice the high-oscillatory behaviour caused by the dead-time mismatch, especially in the temperature output T, for both the optimal and NLFSP predictors.

Using the procedure presented in Section 5.4.5, the disturbance simulation data were analyzed to estimate the frequency domain characteristics of $\mathbf{w}$, which is shown in Figure 57. Note that in this case, there are peaks near 3 [rad/s]. The following prediction error filter were designed to attenuate these frequency components::

$$\mathbf{F_{r1}}(z) = F_{r1}(z)\mathbf{I} = \frac{0.1}{z - 0.9}\mathbf{I}.$$

The simulation results for this case are also presented in Figures 55 and 56. The disturbances $\mathbf{w}$ and $\widetilde{\mathbf{w}}$ plots were again split before and after the external perturbation is applied. The bounds on each disturbance are presented in Table 2. Notice how the bound on $\widetilde{\mathbf{w}}$ is smaller than the one on $\mathbf{w}$ for the NLFSP with filter $\mathbf{F_{r1}}(z)$, which results in a smoother closed-loop response, which indicates a more robust closed-loop system. As stated before, the bound on $\widetilde{\mathbf{w}}$ is smaller because $\mathbf{F_{r1}}(z)$ is a low-pass filter which diminishes the effects of medium and high-frequency signals, which are exactly the type of signals introduced by a dead-time mismatch.

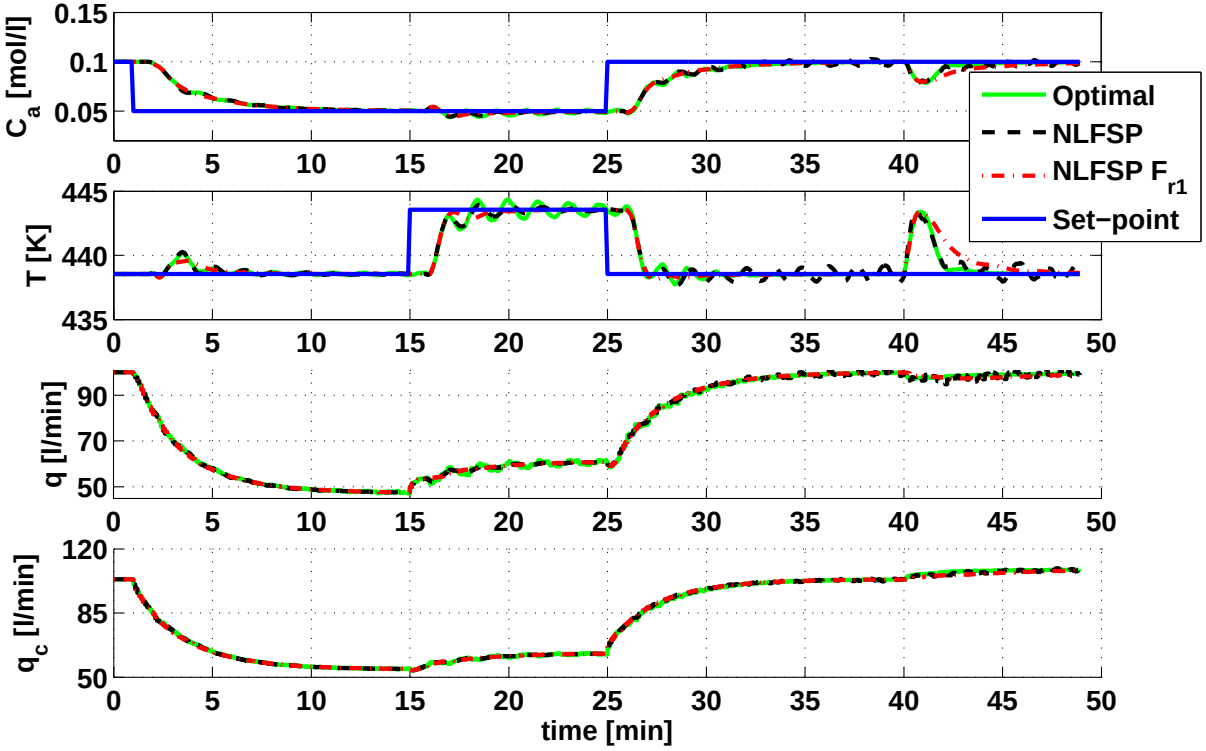

Figure 55 – Closed-loop response of the MIMO CSTR with dead-time mismatch.

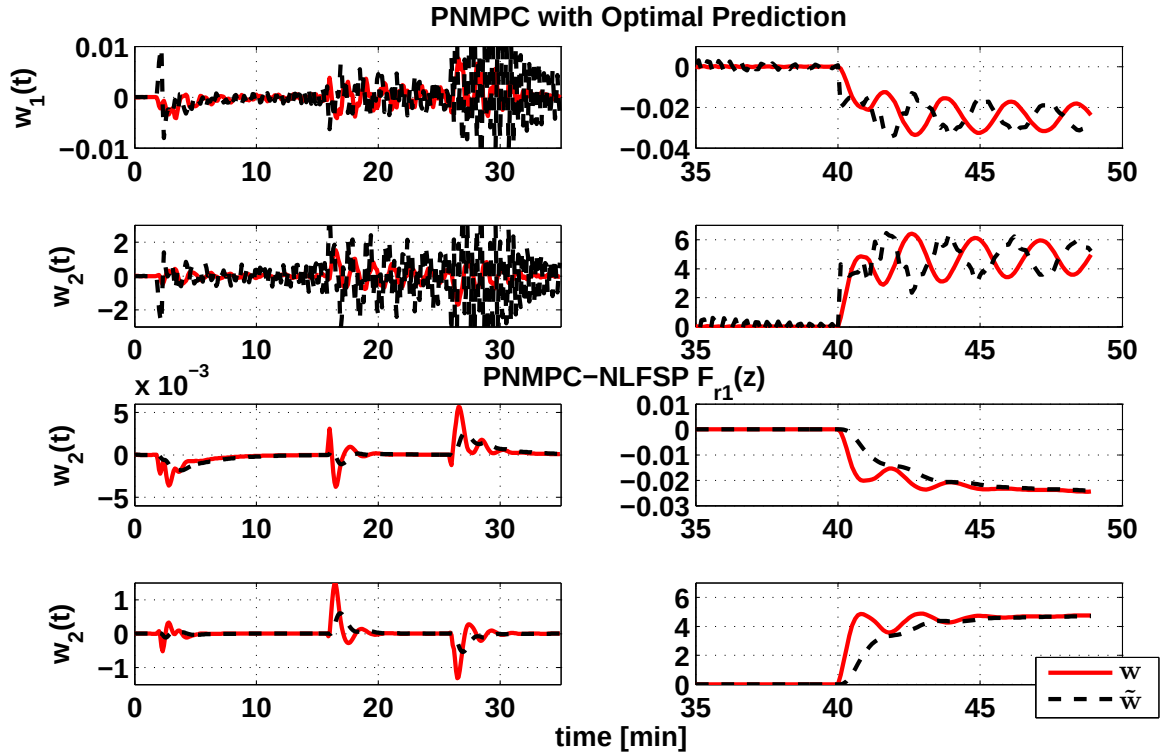

Figure 56 – Disturbances of the closed-loop MIMO CSTR with dead-time mismatch.

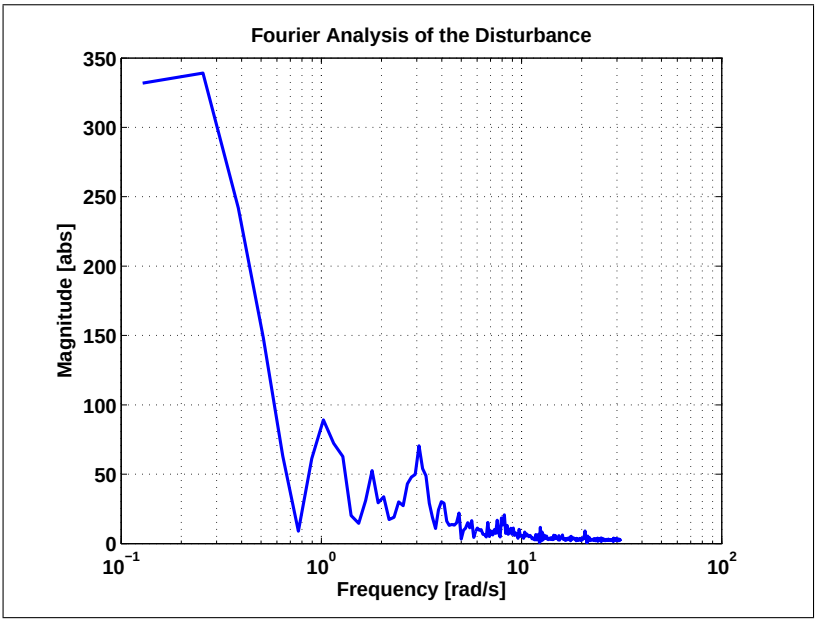

Figure 57 – Frequency domain characteristics of $\mathbf{w}$ obtained through Fourier analysis of simulation data.

6.8 FINAL COMMENTS

In this chapter a Nonlinear Filtered Smith Predictor (NLFSP) for stable nonlinear systems was proposed following the ideas of the linear FSP. However, it was shown that using the usual system description, where the disturbances are applied to the states, it is not possible to extend the properties of the linear FSP to the stable nonlinear case. Hence, an alternative system description was proposed where the disturbances are represented as output disturbances, which permitted to prove that the robustness properties of the linear FSP remain true for stable nonlinear systems, i.e., if the NLFSP is used, the robustness is not dependent on the nominal dead time and, if the prediction filter is chosen adequately, the closed-loop robustness can be improved further.

The same remarks made for the NLFSP for systems with input nonlinearities in Section 5.7 are valid here. The NLFSP properties discussed here are valid with controller, but they are specially interes-

ting in the MPC context, as demonstrated with the examples of this chapter.

7 CONCLUSIONS AND FUTURE WORK

This document presented the results concerning the thesis titled "Predictor-Based Robust control of Dead-Time Processes". The first contribution was the Filtered Dynamic Matrix Control (FDMC), which makes a small modification on the classical DMC algorithm that provides an additional tuning parameter that can be used to improve the closed-loop disturbance rejection and/or increase the closed-loop robustness. As described in Chapter 4, the modification consists of filtering the prediction error. Tuning guidelines were also given to adjust the closed-loop properties of the system.

Since the DMC is still widely used in the industry, this simple modification can easily help improve the production quality. Regarding the FDMC, the following publications were made:

1. Daniel M. Lima, Julio E. Normey-Rico, Agustinho Plucênio, Tito L. M. Santos, Marcos V. C. Gomes. Improving Robustness and Disturbance Rejection Performance with Industrial MPC. XX Congresso Brasileiro de Automática, 2014, Belo Horizonte, Brazil [90].

2. Daniel M. Lima, Tito L. M. Santos, Julio E. Normey-Rico. Filtered Dynamic Matrix Control Applied to a Solar Collector Field. The 6th International Renewable Energy Conference, 2015, Sousse, Tunisia [110].

The main contribution of this thesis is the Nonlinear Filtered Smith Predictor (NLFSP), in which the FSP theory was applied to nonlinear system. In Chapter 5, mathematical proofs were provided which indicated that the linear FSP advantages were still true for systems with input nonlinearities, i.e., the robustness of the closed-loop system is not dependent on the nominal dead time and a filter can be used to improve robustness and disturbance rejection.

In Chapter 6 the NLFSP for general nonlinear systems were discussed. In this case, the robustness properties of the linear FSP still holds. However, the NLFSP can only be applied near stable equilibrium points, and the disturbance rejection properties are still being researched.

The NLFSP can be used with any kind of controller, linear or nonlinear. Nonetheless, as was demonstrated throughout this thesis, it has a particular interest in the MPC context. Various MPC algorithms

use an implicit optimal predictor, which is not the best option when robustness is considered, as proven in Chapters 5 and 6. Substituting this implicit prediction by the NLFSP can lead to a more robust closed-loop system that is easier to design. Also, tuning guidelines were provided for robustness and disturbance rejection.

Regarding the NLFSP, the following publications were made:

1. Daniel M. Lima, Tito L. M. Santos, Julio E. Normey-Rico. Robust Nonlinear Predictor for Dead Time Systems with Input Nonlinearities, Journal of Process Control, Volume 27, March 2015 [103].

2. Daniel M. Lima, Julio E. Normey-Rico, Titlo L. M. Santos. Robust delay compensation for MPC for systems with input nonlinearities and multiple dead times. 5th IFAC Conference on Nonlinear Model Predictive Control (NMPC"15), September 2015, Seville, Spain (Submitted for publishing).

3. Daniel M. Lima, Tito L. M. Santos, Julio E. Normey-Rico. A Robust predictor for nonlinear systems with dead time, 54th Conference on Decision and Control, December 2015, Osaka, Japan (Submitted for publishing).

7.1 FUTURE WORK

It must be noted that this thesis is still an ongoing work, hence, there are topics that need further research. The main ones are listed below:

1. Comparison between FDMC and other modifications of MPC algorithms. For example, in [72], it was suggested the use of a Kalman filter as the disturbance estimator. A comparison between the two should be made to evaluate advantages/disadvantages of each modification. Also, an evaluation of the FDMC properties if the controller is under constraints.

2. Research of other nonlinear predictor structures so as to make a broader comparison between the proposed NLFSP and other techniques, and not only with the optimal predictor.

3. Extension of the NLFSP to the general nonlinear systems case, i.e., to also consider unstable equilibrium points, and the tuning of the prediction filter to improve disturbance rejection.

4. More application, with experimental results, of NMPC algorithms with the NLFSP, to further corroborate the results presented here.

5. In this thesis, the examples and case-studies were modeled in state-space in a way that the states were simply the delayed outputs, hence they were all observable. However, this is not always the case. When some of states are not observable, it is necessary to use state observers, which changes the closed-loop behaviour. Thus, an interesting topic to research is the robustness properties of the system if a predictor and an observer structure are used.

REFERENCES

1 FLESCH, R. C. et al. Unified approach for minimal output dead time compensation in MIMO processes. *Journal of Process Control*, v. 21, n. 7, p. 1080 – 1091, 2011.

2 CAMACHO, E.; BORDONS, C. *Model Predictive Control*. [S.l.]: Springer, Berlin, 2004.

3 NORMEY-RICO, J.; CAMACHO, E. *Control of Dead-Time Processes*. [S.l.]: Springer, 2007. (Advanced Textbooks in Control and Signal Processing).

4 PALMOR, Z. J. Time-delay compensation & Smith predictor and its modifications. In: LEVINE, W. S. (Ed.). *The Control Handbook*. [S.l.]: CRC and IEEE Press, 1996. cap. 10.8, p. 224–237.

5 SMITH, O. J. M. Closed control of loops with dead-time. *Chemical Engineering Progress*, v. 53, n. 5, p. 217–219, 1957.

6 SANTOS, T. L. M. et al. On the prediction error of dead-time compensation control for constrained nonlinear systems. In: *European Control Conference (ECC14), Strasbourg, France*. [S.l.: s.n.], 2014. p. Accepted for publishing.

7 ALBERTOS, P.; GARCíA, P. Robust control design for long time-delay systems. *Journal of Process Control*, v. 19, n. 10, p. 1640 – 1648, 2009. ISSN 0959-1524.

8 SANTOS, T. L. M. *Contribuições para o Controle Preditivo com Compensação de Atraso Robusta*. Tese (Doutorado) — Universidade Federal de Santa Catarina, 2011.

9 QIN, S.; BADGWELL, T. A. A survey of industrial model predictive control technology. *Control Engineering Practice*, v. 11, n. 7, p. 733 – 764, 2003.

10 MAYNE, D.; DONA, J. D.; GOODWIN, G. Improved stabilising conditions for model predictive control. In: *Decision and Control, 2000. Proceedings of the 39th IEEE Conference on*. [S.l.: s.n.], 2000. v. 1, p. 172–177 vol.1.

11 EVANS, M.; CANNON, M.; KOUVARITAKIS, B. Robust MPC for linear systems with bounded multiplicative uncertainty. In: *2012 IEEE 51st Annual Conference on Decision and Control(CDC)*. [S.l.: s.n.], 2012. p. 248–253.

12 MARRUEDO, D.; ALAMO, T.; CAMACHO, E. Input-to-state stable MPC for constrained discrete-time nonlinear systems with bounded additive uncertainties. In: *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*. [S.l.: s.n.], 2002. v. 4, p. 4619–4624 vol.4.

13 CUTLER, C.; RAMAKER, B. Dynamic matrix control–a computer control algorithm. 1979.

14 LUNDSTRÖM, P. et al. Limitations of dynamic matrix control. *Computers & chemical engineering*, Elsevier, v. 19, n. 4, p. 409–421, 1995.

15 ROSSITER, J. *Model-Based Predictive Control: A Practical Approach*. [S.l.]: CRC Press, 2003. (CRC Press Control Series).

16 JØRGENSEN, J.; HUUSOM, J.; RAWLINGS, J. Finite horizon MPC for systems in innovation form. In: *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*. [S.l.: s.n.], 2011. p. 1896–1903.

17 LEE, J.; YU, Z. Tuning of model predictive controllers for robust performance. *Computers & Chemical Engineering*, v. 18, n. 1, p. 15 – 37, 1994. An International Journal of Computer Applications in Chemical Engineering.

18 KOUVARITAKIS, B.; CANNON, M.; ENGINEERS, I. of E. *Non-Linear Predictive Control: Theory and Practice*. [S.l.]: Institution of Engineering and Technology, 2001. (IEE Monographs).

19 GRÜNE, L.; PANNEK, J. *Nonlinear model predictive control*. [S.l.]: Springer, 2011.

20 MAGNI, L.; RAIMONDO, D. M.; ALLGÖWER, F. *Nonlinear model predictive control*. [S.l.]: Springer, 2009.

21 SANTOS, T. et al. Sobre o efeito do atraso nominal na robustez de estratégias com compensação de atraso. In: *XIX Congresso Brasileiro de Automática*. [S.l.: s.n.], 2012.

22 THOMÉ, F.; SANTOS, T. L.; NORMEY-RICO, J. E. Compensador de atraso modificado para sistemas não-lineares com tempo morto. *XI SBAI - Simpósio Brasileiro de Automamção Inteligente*, 2013.

23 THOMÉ, F. F. *Controle preditivo robusto de sistemas não lineares com atraso*. Dissertação (Mestrado) — Universidade Federal de Santa Catarina, Centro Tecnológico, Programa de Pós-Graduação em Engenharia de Automação e Sistemas, 2013.

24 BEKIARIS-LIBERIS, N.; KRSTIC, M. Robustness of nonlinear predictor feedback laws to time- and state-dependent delay perturbations. *Automatica*, v. 49, n. 6, p. 1576 – 1590, 2013.

25 KRSTIC, M. Input delay compensation for forward complete and strict-feedforward nonlinear systems. *Automatic Control, IEEE Transactions on*, v. 55, n. 2, p. 287–303, Feb 2010.

26 KARAFYLLIS, I.; KRSTIC, M. Stabilization of nonlinear delay systems using approximate predictors and high-gain observers. *Automatica*, Pergamon Press, Inc., v. 49, n. 12, p. 3623–3631, dez. 2013.

27 KRAVARIS, C.; WRIGHT, R. A. Deadtime compensation for nonlinear processes. *AIChE Journal*, Wiley Online Library, v. 35, n. 9, p. 1535–1542, 1989.

28 HENSON, M. A.; SEBORG, D. E. Time delay compensation for nonlinear processes. *Industrial & Engineering Chemistry Research*, v. 33, n. 6, p. 1493–1500, 1994.

29 JIN-QUAN, H.; LEWIS, F. L. Neural-network predictive control for nonlinear dynamic systems with time-delay. *Neural Networks, IEEE Transactions on*, IEEE, v. 14, n. 2, p. 377–389, 2003.

30 PAGANO, D.; NORMEY-RICO, J.; FRANCO, A. Stability analysis of a modified smith predictor for integrative plants with dead-time uncertainties and saturations. In: *Decision and Control, 2001. Proceedings of the 40th IEEE Conference on*. [S.l.: s.n.], 2001. v. 2, p. 1855–1860.

31 SBARCIOG, M. et al. Nonlinear predictive control of processes with variable time delay. a temperature control case study. In: IEEE. *Control Applications, 2008. CCA 2008. IEEE International Conference on*. [S.l.], 2008. p. 1001–1006.

32 GÁLVEZ-CARRILLO, M.; KEYSER, R. D.; IONESCU, C. Nonlinear predictive control with dead-time compensator: Application to a solar power plant. *Solar energy*, Elsevier, v. 83, n. 5, p. 743–752, 2009.

33 TORRICO, B. C. et al. Robust nonlinear predictive control applied to a solar collector field in a solar desalination plant. *Control Systems Technology, IEEE Transactions on*, IEEE, v. 18, n. 6, p. 1430–1439, 2010.

34 NORMEY-RICO, J. E.; CAMACHO, E. F. Unified approach for robust dead-time compensator design. *Journal of Process Control*, v. 19, n. 1, p. 38 – 47, 2009. ISSN 0959-1524.

35 NORMEY-RICO, J.; CAMACHO, E. A unified approach to design dead-time compensators for stable and integrative processes with dead-time. *IEEE Trans. on Automatic Control*, v. 47, n. 2, p. 299–305, February 2002.

36 ZHONG, Q.-C.; NORMEY-RICO, J. Control of integral processes with dead time. Part 1: Disturbance observer-based 2DOF control scheme. *Control Theory and Applications, IEE Proceedings*, v. 149, n. 4, p. 285–290, July 2002.

37 ZHONG, Q.-C.; MIRKIN, L. Control of integral processes with dead time. Part 2: Quantitative analysis. *IEE Proc. Control Theory Appl.*, v. 149(4), p. 291–296, 2002.

38 ZHONG, Q.-C.; LI, H. Control of integral processes with dead time. Part 3: Dead-beat disturbance response. *IEEE Trans. on Automatic Control*, v. 48, n. 1, p. 153 –159, 2003.

39 ASTRÖM, K.; HANG, C.; LIM, B. A new Smith predictor for controlling a process with a integrator and long dead time. *IEEE Trans. on Automatic Control*, v. 39, n. 2, p. 343–345, 1994.

40 MATASEK, M.; MICIĆ, A. A modified Smith predictor for controlling a process with a integrator and long dead time. *IEEE Trans. on Automatic Control*, v. 41, n. 8, p. 1199–1203, August 1996.

41 MATASEK, M.; MICIĆ, A. On the modified Smith predictor for controlling a process with a integrator and long dead-time. *IEEE Trans. on Automatic Control*, v. 44, n. 8, p. 1603–1606, 1999.

42 KWAK, H.; WHAN, S.; LEE, I.-B. Modified Smith predictor for integrating processes: comparisons and proposition. *Ind. Eng. Chem. Res.*, v. 40, p. 1500–1506, 2001.

43 KAYA, I. Obtaining controller parameters for a new PI-PD Smith predictor using auto tuning. *Journal of Process Control*, v. 13, p. 465–472, 2003.

44 CHIEN, I.-L.; PENG, S.; LIU, J. Simple control method for integrating processes with long deadtime. *Journal of Process Control*, v. 12, n. 3, p. 391–404, 2002.

45 HANG, C.; WANG, Q.-G.; YANG, X.-P. A modified Smith predictor for a process with an integrator and long dead time. *Ind. Eng. Chem. Res.*, v. 42, p. 484–489, 2003.

46 KWAK, I. et al. A modified Smith predictor with a new structure for unstable processes. *Ind. Eng. Chem. Res.*, v. 38, p. 405, 1999.

47 TAN, W.; MARQUEZ, I.; CHEN, T. IMC design for ustable processes with time delay. *Int. Journal of Control*, v. 45, p. 8291, 2005.

48 LIU, T. et al. New modified Smith predictor scheme for integrating and unstable processes with time delay. *IEE Proc. Control Theory Appl.*, v. 152(2), p. 238–246, 2005.

49 LIU, T.; ZHANG, W.; GU, D. Analytical design of two-degree-of-fredom control scheme for open-loop unstable processes with time delay. *Journal of Process Control*, v. 15(5), p. 559, 2005.

50 LU, X. et al. A double two-degree-of-freedom control scheme for improved control of unstable delay processes. *Journal of Process Control*, v. 15(5), p. 605–614, 2005.

51 RAO, A.; CHIDAMBARAM, M. Enhanced Smith predictor for unstable processes with time delay. *Ind. Eng. Chem. Res.*, v. 44, p. 8291–8299, 2005.

52 RAO, A.; CHIDAMBARAM, M. Simple Analytical Design of Modified Smith Predictor with Improved Performance for Unstable First-Order Plus Time Delay (FOPTD) Processes. *Ind. Eng. Chem. Res.*, v. 46, p. 4561–4571, 2007.

53 PALMOR, Z.; HALEVI, Y. Robustness properties of sampled-data systems with dead time compensators. *Automatica*, v. 26, p. 637–640, 1990.

54 GUO, S.; WANG, W.; SHIEH, L. Discretisation of two degree-of-freedom controller and system with state, and output delays. *IEE Proceedings. Control Theory and Applications*, v. 147, n. 1, p. 87–96, January 2000.

55 TORRICO, B.; NORMEY-RICO, J. 2DOF discrete dead-time compensators for stable and integrative processes with dead time. *Journal of Process Control*, v. 15, p. 341–352, 2005.

56 NORMEY-RICO, J.; BORDONS, C.; CAMACHO, E. Improving the robustness of dead-time compensating PI controllers. *Control Engineering Practice*, v. 5, n. 6, p. 801–810, 1997.

57 ASTRÖM, K.; WITTERMARK, B. *Computed Controlled Systems*. [S.l.]: Prentice Hall, Englewood Cliffs, NJ, 1984.

58 MORARI, M.; ZAFIRIOU, E. *Robust Process Control*. [S.l.]: Prentice Hall, Englewood Cliffs, 1989.

59 SANTOS, T. L. M.; BOTURA, P. E. A.; NORMEY-RICO, J. E. Dealing with noise in unstable dead-time process control. *Journal of Process Control*, v. 20, n. 7, p. 840 – 847, 2010.

60 FLESCH, R. C.; SANTOS, T. L.; NORMEY-RICO, J. E. Unified approach for minimal output dead time compensation in mimo non-square processes. In: *CDC*. [S.l.: s.n.], 2012. p. 2376–2381.

61 SANTOS, T. L.; FLESCH, R. C.; NORMEY-RICO, J. E. On the filtered Smith predictor for MIMO processes with multiple time delays. *Journal of Process Control*, v. 24, n. 4, p. 383 – 400, 2014. ISSN 0959-1524.

62 SKOGESTAD, S.; POSTLETHWAITE, I. *Multivariable feedback control: Analysis and design*. [S.l.]: John Wiley and Sons, 2001.

63 WOOD, R. K.; BERRY, M. W. Terminal composition control of a binary distillation column. *Chem Eng Sci*, v. 28, n. 9, p. 1707–1717, 1973.

64 LELIĆ, M.; ZARROP, M. Generalized pole-placement self-tuning controller part 1, basic algorithm. *International Journal of Control*, Taylor & Francis, v. 46, n. 2, p. 547–568, 1987.

65 LINKERS, D.; MAHFONF, M. Advances in model-based predictive control. In: _____. [S.l.]: Oxford University Press, 1994. cap. Generalized Predictive Control in Clinical Anaesthesia.

66 CLARKE, D. Application of generalized predictive control to industrial processes. *IEEE Control Systems Magazine*, v. 122, p. 49–55, 1988.

67 RICHALET, J. Industrial applications of model based predictive control. *Automatica*, v. 29, n. 5, p. 1251–1274, 1993.

68 RICHALET, J. et al. Model predictive heuristic control: Application to industrial processes. *Automatica*, v. 14, n. 2, p. 413–428, 1978.

69 TAKATSU, H.; ITOH, T.; ARAKI, M. Future needs for the control theory in industries. Report and topics of the control technology survey in Japanese industry. *Journal of Process Control*, v. 8, p. 369–374, 1998.

70 ALESSIO, A.; BEMPORAD, A. A survey on explicit model predictive control. In: MAGNI, L.; RAIMONDO, D.; ALLGöWER, F. (Ed.). *Nonlinear Model Predictive Control*. [S.l.]: Springer Berlin Heidelberg, 2009, (Lecture Notes in Control and Information Sciences, v. 384). p. 345–369.

71 BEMPORAD, A. et al. The explicit linear quadratic regulator for constrained systems. *Automatica*, Elsevier, v. 38, n. 1, p. 3–20, 2002.

72 LEE, J. H.; MORARI, M.; GARCIA, C. E. State-space interpretation of model predictive control. *Automatica*, Elsevier, v. 30, n. 4, p. 707–717, 1994.

73 DATTA, A.; OCHOA, J. Adaptive internal model control: Design and stability analysis. *Automatica*, v. 32, n. 2, p. 261–266, 1996.

74 CLARKE, D. W.; MOHTADI, C.; TUFFS, P. S. Generalized predictive control - Part I. The basic algorithm. *Automatica*, p. 137–148, 1987.

75 KEYSER, R. M. C. De; CUAWENBERGHE, A. R. Van. Extended prediction self-adaptive control. In: *IFAC Symposium on Identification and System Parameter Estimation*. [S.l.: s.n.], 1985. p. 1317–13228.

76 CANNON, M. Efficient nonlinear model predictive control algorithms. *Annual Reviews in Control*, Elsevier, v. 28, n. 2, p. 229–237, 2004.

77 DOYLE, F. J.; OGUNNAIKE, B. A.; PEARSON, R. K. Nonlinear model-based control using second-order volterra models. *Automatica*, v. 31, n. 5, p. 697 – 714, 1995. ISSN 0005-1098.

78 SANTOS, J. E. S. dos. *Controle Preditivo Não Linear para Sistemas de Hammerstein*. Tese (Doutorado) — Universidade Federal de Santa Catarina, Centro Tecnológico, 2005.

79 BLOEMEN, H.; BOOM, T. Van den; VERBRUGGEN, H. Model-based predictive control for hammerstein systems. In: IEEE. *Decision and Control, 2000. Proceedings of the 39th IEEE Conference on*. [S.l.], 2000. v. 5, p. 4963–4968.

80 MANTHANWAR, A. M.; SAKIZLIS, V.; PISTIKOPOULOS, E. Robust parametric predictive control design for polytopically uncertain systems. In: IEEE. *American Control Conference, 2005. Proceedings of the 2005*. [S.l.], 2005. p. 3994–3999.

81 LU, Y.; ARKUN, Y. A scheduling quasi-minmax MPC for LPV systems. In: IEEE. *American Control Conference, 1999. Proceedings of the 1999*. [S.l.], 1999. v. 4, p. 2272–2276.

82 PLUCÊNIO, A. *Development of Nonlinear Control Techniques for the lifting of Multiphase Fluids (Desenvolvimento de Técnicas de Controle Não Linear para Elevação de Fluídos Multifásicos)*. Tese (Doutorado) — Universidade Federal de Santa Catarina, Centro Tecnológico, Programa de Pós-Graduação em Engenharia de Automação e Sistemas, 2010.

83 PLUCÊNIO, A. et al. A practical approach to predictive control for nonlinear processes. In: *NOLCOS 2007 - 7th IFAC Symposium on Nonlinear Control Systems*. [S.l.: s.n.], 2007.

84 MORARI, M. *Advances in Model-Based Predictive Control, chapter Model Predictive Control: Multivariable Control Technique of Choice in the 1990s*. [S.l.]: Oxford University Press, 1994.

85 CLARKE, D. W.; MOHTADI, C. Properties of generalized predictive control. *Automatica*, Elsevier, v. 25, n. 6, p. 859–875, 1989.

86 ROBINSON, B.; CLARKE, D. Robustness effects of a prefilter in generalised predictive control. In: IET. *IEE Proceedings D (Control Theory and Applications)*. [S.l.], 1991. v. 138, n. 1, p. 2–8.

87 YOON, T.-W.; CLARKE, D. W. Observer design in receding-horizon predictive control. *International Journal of Control*, Taylor & Francis, v. 61, n. 1, p. 171–191, 1995.

88 ANSAY, P.; WERTZ, V. Model uncertainties in gpc: A systematic two-step design. In: *Proc. of the ECC*. [S.l.: s.n.], 1997. v. 97.

89 NORMEY-RICO, J.; CAMACHO, E. Multivariable generalised predictive controller based on the Smith predictor. *IEE Proceedings-Control Theory and Applications*, IET, v. 147, n. 5, p. 538–546, 2000.

90 LIMA, D. M. et al. Improving Robustness and Disturbance Rejection Performance with Industrial MPC. In: *XX Congresso Brasileiro de Automática*. [S.l.: s.n.], 2014. p. 3229–3236.

91 MORARI, M.; LEE, J.; GARCIA, C. *Model Predictive Control*. [S.l.]: Prentice Hall PTR, 2002.

92 PEARSON, R. Selecting nonlinear model structures for computer control. *Journal of Process Control*, v. 13, n. 1, p. 1 – 26, 2003.

93 MANER, B. et al. A nonlinear model predictive control scheme using second order volterra models. In: *American Control Conference, 1994*. [S.l.: s.n.], 1994. v. 3, p. 3253–3257 vol.3.

94 DOYLE, F.; PEARSON, R.; OGUNNAIKE, B. *Identification and Control Using Volterra Models*. [S.l.]: Springer London, 2002. (Communications and Control Engineering).

95 GIANNAKIS, G. B.; SERPEDIN, E. A bibliography on nonlinear system identification. *Signal Processing*, v. 81, n. 3, p. 533 – 580, 2001. Special section on Digital Signal Processing for Multimedia.

96 HABER, R. Predictive control of nonlinear dynamic processes. *Applied Mathematics and Computation*, v. 70, p. 169 – 184, 1995. Dynamics and Control.

97 ALONGE, F. et al. Identification of nonlinear systems described by hammerstein models. In: *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*. [S.l.: s.n.], 2003. v. 4, p. 3990–3995 vol.4.

98 FRUZZETTI, K.; PALAZOGLU, A.; MCDONALD, K. Nonlinear model predictive control using hammerstein models. *Journal of Process Control*, v. 7, n. 1, p. 31–41, 1997.

99 JIANG, Z.-P.; WANG, Y. Input-to-state stability for discrete-time nonlinear systems. *Automatica*, v. 37, n. 6, p. 857 – 869, 2001.

100 LIMON, D. et al. Input-to-state stability: A unifying framework for robust model predictive control. In: MAGNI, L.; RAIMONDO, D.; ALLGöWER, F. (Ed.). *Nonlinear Model Predictive Control*. [S.l.]: Springer Berlin Heidelberg, 2009, (Lecture Notes in Control and Information Sciences, v. 384). p. 1–26.

101 YOON, T.; CLARKE, D. *Advances in Model-Based Predictive Control, Towards Robust Adaptive Predictive Control*. [S.l.]: Oxford University Press, Oxford, UK, 1994.

102 YOON, T.-W.; CLARKE, D. W. Observer design in receding-horizon predictive control. *International Journal of Control*, Taylor & Francis, v. 61, n. 1, p. 171–191, 1995.

103 LIMA, D. M.; SANTOS, T. L. M.; NORMEY-RICO, J. E. Robust nonlinear predictor for dead-time systems with input nonlinearities. *Journal of Process Control*, v. 27, n. 0, p. 1 – 14, 2015. ISSN 0959-1524.

104 NORMEY-RICO, J. E.; GARCIA, P.; GONZALEZ, A. Robust stability analysis of filtered Smith predictor for time-varying delay processes. *Journal of Process Control*, v. 22, n. 10, p. 1975 – 1984, 2012. ISSN 0959-1524.

105 BEQUETTE, B. *Process dynamics: modeling, analysis, and simulation*. [S.l.]: Prentice Hall PTR, 1998. (Prentice-Hall international series in the physical and chemical engineering sciences).

106 ENGELL, S.; KLATT, K.-U. Nonlinear control of a non-minimum-phase cstr. In: *American Control Conference, 1993*. [S.l.: s.n.], 1993. p. 2941–2945.

107 VUSSE, J. van de. Plug-flow type reactor versus tank reactor. *Chemical Engineering Science*, v. 19, n. 12, p. 994 – 996, 1964.

108 LAZAR, M. *Nonlinear controller based on the EPSAC approach*. Dissertação (Mestrado) — University of Ghent, Flanders, Belgium, 2001.

109 DU, J.; JOHANSEN, T. A. A gap metric based weighting method for multimodel predictive control of {MIMO} nonlinear systems. *Journal of Process Control*, v. 24, n. 9, p. 1346 – 1357, 2014. ISSN 0959-1524.

110 LIMA, D. M.; NORMEY-RICO, J. E.; SANTOS, T. L. M. Filtered dynamic matrix control applied to a solar collector field. In: IEEE. *Renewable Energy Congress (IREC), 2015 6th International*. [S.l.], 2015. p. 1–6.

111 OPPENHEIM, A. V.; WILLSKY, A. S.; NAWAB, S. H. *Signals & Systems*. [S.l.]: Prentice Hall, 1996. (Prentice Hall Signal Processing Series).

112 LIU, Y. *Analysis and Design of Systems with a Non-Negative Impulse Response*. Tese (Doutorado) — University of Notre Dame, 2011.

113 SONTAG, E. D. Smooth stabilization implies coprime factorization. *Automatic Control, IEEE Transactions on*, IEEE, v. 34, n. 4, p. 435–443, 1989.

114 SONTAG, E. D. Further facts about input to state stabilization. *IEEE Transactions on Automatic Control*, v. 35, n. 4, p. 473–476, 1990.

Appendices

A PROOFS OF THEOREMS

A.1 PROOF OF THEOREM 5.1

First, consider the following Lemma:

Lemma A.1. *Given the model described by Eq. (5.10) that is used by the optimal predictor, the optimal prediction of the ith group of system states at time $t + d_i$, with the information up to time t, is*

$$\mathbf{x_i}(t + d_i|t) = \mathbf{A_i}^{d_i}\mathbf{x_i}(t) + \sum_{j=1}^{d_i} \mathbf{A_i}^{j-1} g_i(\mathbf{u}(t-j)) + \sum_{j=1}^{d_i} \mathbf{A_i}^{j-1}\mathbf{w_i}(t-1). \tag{A.1}$$

Proof. Multiplying both sides of Eq. (5.10) by Δ

$$\begin{aligned}\mathbf{x}(t+1)\Delta =&(\mathbf{A}\mathbf{x}(t) + g(\mathbf{u}(t-D)))\Delta + \mathbf{e}(t)\\ \mathbf{x}(t+1) - \mathbf{x}(t) =&\mathbf{A}\mathbf{x}(t) + g(\mathbf{u}(t-D))\\ &- (\mathbf{A}\mathbf{x}(t-1) + g(\mathbf{u}(t-D-1))) + \mathbf{e}(t).\end{aligned}$$

Considering that the expected value of $\mathbf{e}(t+i) = 0$, $\forall i \geq 0$, the prediction of the states at time $t+1$, given the information at instant t, is

$$\begin{aligned}\mathbf{x}(t+1|t) =&\mathbf{A}\mathbf{x}(t) + g(\mathbf{u}(t-D))\\ &+ (\mathbf{x}(t) - \mathbf{A}\mathbf{x}(t-1) - g(\mathbf{u}(t-D-1))).\end{aligned} \tag{A.2}$$

By Eq. (5.5), the last term of Eq. (A.2) is

$$\mathbf{x}(t) - \mathbf{A}\mathbf{x}(t-1) - g(\mathbf{u}(t-D-1)) = \mathbf{w}(t-1),$$

i.e., the optimal predictor assumes that $\mathbf{w}(t) = \mathbf{w}(t-1)$ and, as will be shown in the sequel, that this disturbance is constant over the prediction horizon. Using this equality on Eq. (A.2)

$$\mathbf{x}(t+1|t) = \mathbf{A}\mathbf{x}(t) + g(\mathbf{u}(t-D)) + \mathbf{w}(t-1). \tag{A.3}$$

The plant state at $t+2$ is

$$\mathbf{x}(t+2) = \mathbf{A}\mathbf{x}(t+1) + g(\mathbf{u}(t-D+1)) + \frac{\mathbf{e}(t+1)}{\Delta},$$

Applying the same steps used to obtain $\mathbf{x}(t+1|t)$

$$\begin{aligned}\mathbf{x}(t+2|t) - \mathbf{x}(t+1|t) =& \mathbf{A}\mathbf{x}(t+1|t) + g(\mathbf{u}(t-D+1)) \\ & - \mathbf{A}\mathbf{x}(t) - g(\mathbf{u}(t-D)),\end{aligned}$$

and using Eq. (A.3) it can be shown that the prediction at $t+2$ is

$$\begin{aligned}\mathbf{x}(t+2|t) =& \mathbf{A}^2\mathbf{x}(t) + \mathbf{A}g(\mathbf{u}(t-D)) + g(\mathbf{u}(t-D+1)) \\ & + \mathbf{A}\mathbf{w}(t-1) + \mathbf{w}(t-1).\end{aligned} \tag{A.4}$$

By inspection of Eq. (A.3) and Eq. (A.4),

$$\begin{aligned}\mathbf{x}(t+k|t) =& \mathbf{A}^k\mathbf{x}(t) + \sum_{j=1}^{k-1} \mathbf{A}^{k-1-j} g(\mathbf{u}(t-D+j)) \\ & + \sum_{j=0}^{k-1} \mathbf{A}^{k-1-j}\mathbf{w}(t-1).\end{aligned}$$

Since the matrix $\mathbf{A}$ is block diagonal, so is $\mathbf{A}^l$, $\forall l$. Thus, the prediction at time $t+k$ of each ith group of states can be written as

$$\begin{aligned}\mathbf{x_i}(t+k|t) =& {\mathbf{A_i}}^k\mathbf{x_i}(t) + \sum_{j=1}^{k-1} {\mathbf{A_i}}^{k-1-j} g_i(\mathbf{u}(t-d_i+j)) \\ & + \sum_{j=0}^{k-1} {\mathbf{A_i}}^{k-1-j}\mathbf{w_i}(t-1).\end{aligned}$$

Then, the prediction of the ith group of states at time $t+d_i$ is, after some rearrangements,

$$\mathbf{x_i}(t+d_i|t) = {\mathbf{A_i}}^{d_i}\mathbf{x_i}(t) + \sum_{j=1}^{d_i} \mathbf{A}^{j-1} g_i(\mathbf{u}(t-j)) + \sum_{j=0}^{d_i} {\mathbf{A_i}}^{j-1}\mathbf{w_i}(t-1).$$

□

Now, the proof of Theorem 5.1:

Theorem. *Consider a dead-time system with input nonlinearities described by Eq. (5.4), and the dead-time-free model given by Eq. (5.8). If the optimal predictor, which utilizes the model described by Eq. (5.10), is used, the equivalent disturbance is*

$$\widetilde{\mathbf{w}}(t) = \mathbf{w}(t) + \mathbf{A}'(\mathbf{w}(t) - \mathbf{w}(t-1)), \tag{A.5}$$

with

$$\mathbf{A}' = \begin{bmatrix} \sum_{j=1}^{d_1} \mathbf{A_1}^j & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sum_{j=1}^{d_p} \mathbf{A_p}^j \end{bmatrix}.$$

Proof. Rearranging Eq. (5.8)

$$\widetilde{\mathbf{w}}(t) = \widetilde{\mathbf{x}}(t+1) - \mathbf{A}\widetilde{\mathbf{x}}(t) - g(\mathbf{u}(t)), \tag{A.6}$$

and given that

$$\widetilde{\mathbf{x}}(t) \triangleq \mathbf{x}(t+D|t) = \begin{bmatrix} \mathbf{x}(t+d_1|t) \\ \vdots \\ \mathbf{x}(t+d_p|t) \end{bmatrix},$$

in a way that $\widetilde{\mathbf{x}}(t+1) = \mathbf{x}(t+D+1|t+1)$, using Eq. (A.1) from Lemma A.1 and separating the equivalent disturbance $\widetilde{\mathbf{w}}$ in groups because $\mathbf{A}$ is diagonal results in

$$\begin{aligned} \widetilde{\mathbf{w}}_\mathbf{i}(t) =& \mathbf{A_i}^{d_i}\mathbf{x_i}(t+1) + \sum_{j=1}^{d_i} \mathbf{A_i}^{j-1} g_i(\mathbf{u}(t-j+1)) \\ &+ \sum_{j=1}^{d_i} \mathbf{A_i}^{j-1}\mathbf{w_i}(t) - \mathbf{A_i}^{d_i+1}\mathbf{x_i}(t) - \sum_{j=1}^{d} \mathbf{A_i}^{j} g_i(\mathbf{u}(t-j)) \\ &- \sum_{j=1}^{d_i} \mathbf{A_i}^{j}\mathbf{w_i}(t-1) - g_i(\mathbf{u}(t)). \end{aligned} \tag{A.7}$$

With some manipulations, it is easy to show that the terms in-

volving $\mathbf{u}(t)$ can be reduced to

$$\sum_{j=1}^{d_i} {\mathbf{A_i}}^{j-1} g_i(\mathbf{u}(t-j+1))$$
$$-\sum_{j=1}^{d_i} {\mathbf{A_i}}^{j} g(\mathbf{u}(t-j)) - g_i(\mathbf{u}(t)) = -{\mathbf{A_i}}^{d_i} g_i(\mathbf{u}(t-d_i)),$$

which reduces Eq. (A.7) to

$$\begin{aligned}\widetilde{\mathbf{w}}_\mathbf{i}(t) =& {\mathbf{A_i}}^{d_i}(\mathbf{x_i}(t+1) - \mathbf{A_i}\mathbf{x_i}(t) - g_i(\mathbf{u}(t-d_i))) \\ &+ \sum_{j=1}^{d_i} {\mathbf{A_i}}^{j-1}\mathbf{w_i}(t) - \sum_{j=1}^{d_i} {\mathbf{A_i}}^{j}\mathbf{w_i}(t-1). \end{aligned} \tag{A.8}$$

Applying Eq. (5.5) in Eq. (A.8),

$$\widetilde{\mathbf{w}}_\mathbf{i}(t) = {\mathbf{A_i}}^{d_i}\mathbf{w_i}(t) + \sum_{j=1}^{d_i} {\mathbf{A_i}}^{j-1}\mathbf{w_i}(t) - \sum_{j=1}^{d_i} {\mathbf{A_i}}^{j}\mathbf{w_i}(t-1),$$

rearranging the first summation and taking out its first term

$$\begin{aligned}\widetilde{\mathbf{w}}_\mathbf{i}(t) =& \mathbf{w_i}(t) + \sum_{j=1}^{d_i} {\mathbf{A_i}}^{j}\mathbf{w_i}(t) - \sum_{j=1}^{d_i} {\mathbf{A_i}}^{j}\mathbf{w_i}(t-1) \\ =& \mathbf{w_i}(t) + \sum_{j=1}^{d_i} {\mathbf{A_i}}^{j}(\mathbf{w_i}(t) - \mathbf{w_i}(t-1)) \\ =& \mathbf{w_i}(t) + \mathbf{A'_i}(\mathbf{w_i}(t) - \mathbf{w_i}(t-1)), \end{aligned} \tag{A.9}$$

with $\mathbf{A'_i} = \sum_{j=1}^{d_i} {\mathbf{A_i}}^{j}$.

The total equivalent disturbance vector $\widetilde{\mathbf{w}}(t)$ is then given by

$$\widetilde{\mathbf{w}}(t) = \mathbf{w}(t) + \mathbf{A'}(\mathbf{w}(t) - \mathbf{w}(t-1)),$$

with

$$\mathbf{A'} = \begin{bmatrix} \sum_{j=1}^{d_1} {\mathbf{A_1}}^{j} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sum_{j=1}^{d_p} {\mathbf{A_p}}^{j} \end{bmatrix},$$

which completes the proof.

□

A.2 PROOF OF THEOREM 5.2

Theorem. *Consider a dead-time system with input nonlinearities described by Eq. (5.4), and the dead-time-free model given by Eq. (5.8). If the NLFSP structure is used, the equivalent disturbance is*

$$\widetilde{\mathbf{w}}(t) = \mathbf{F_r}(q)\mathbf{w}(t),$$

with $\mathbf{F_r}(q) = F_r(q)\mathbf{I}$*, where* $F_r(q)$ *is a SISO filter.*

Proof. Using the equivalent dead-time-free model given by Eq. (5.8), and considering the NLFSP described by Eq. (5.13a) and Eq. (5.13b),

$$\begin{aligned}\widetilde{\mathbf{w}}(t) =& \mathbf{x}(t+D+1|t+1) - \mathbf{A}\mathbf{x}(t+D|t) - g(\mathbf{u}(t)) \\ =& \mathbf{x_n}(t+D+1) + \mathbf{F_r}(q)(\mathbf{x}(t+1) - \mathbf{x_n}(t+1)) \\ & - \mathbf{A}\mathbf{x_n}(t+D) - \mathbf{A}\mathbf{F_r}(q)(\mathbf{x}(t) - \mathbf{x_n}(t)) - g(\mathbf{u}(t))\end{aligned}$$

using Eq. (5.13a) to obtain $\mathbf{x_n}(t+1)$ and $\mathbf{x_n}(t+D+1)$, and Eq. (5.4)

$$\begin{aligned}\widetilde{\mathbf{w}}(t) =& \mathbf{A}\mathbf{x_n}(t+D) + g(\mathbf{u}(t)) \\ & + \mathbf{F_r}(q)\left\{\mathbf{A}\mathbf{x}(t) + g(\mathbf{u}(t-D)) + \mathbf{w}(t)\right. \\ & \quad \left. -\mathbf{A}\mathbf{x_n}(t) - g(\mathbf{u}(t-D))\right\} \\ & - \mathbf{A}\mathbf{x_n}(t+D) - \mathbf{A}\mathbf{F_r}(q)(\mathbf{x}(t) - \mathbf{x_n}(t)) - g(\mathbf{u}(t)) \\ =& \left\{\mathbf{F_r}(q)\mathbf{A} - \mathbf{A}\mathbf{F_r}(q)\right\}(\mathbf{x}(t) - \mathbf{x_n}(t)) + \mathbf{F_r}(q)\mathbf{w}(t)\end{aligned}$$

since $\mathbf{F_r}(q) = F_r(q)\mathbf{I}$, $\mathbf{F_r}(q)\mathbf{A} - \mathbf{A}\mathbf{F_r}(q) = F_r(q)(\mathbf{A} - \mathbf{A}) = 0$, hence

$$\widetilde{\mathbf{w}}(t) = \mathbf{F_r}(q)\mathbf{w}(t),$$

which completes the proof[1]. □

[1] If $\mathbf{F_r}(q)$ and $\mathbf{A}$ are block diagonal, and their elements have the same dimension, i.e., $\mathbf{F_{ri}}(q)$ and $\mathbf{A_i}$ are square matrices with dimension n_i, $\mathbf{F_r}(q)\mathbf{A} - \mathbf{A}\mathbf{F_r}(q) = 0$, hence the theorem remains true.

A.3 PROOF OF THEOREM 6.1

Theorem. *Consider an open-loop stable nonlinear dead-time system (Assumption 6.2) described by Eq. (6.2), and the dead-time-free model given by Eq. (6.4), where the function $f(.)$ follows Assumption 6.1. If the NLFSP, which is described by Eq. (6.5), is used, the bound on $\widetilde{\mathbf{w}}$ at time t is*

$$\begin{aligned}|\widetilde{\mathbf{w}}(t)| \leq & \sigma_x(|\mathbf{F_r}(q)(\mathbf{x}(t)-\mathbf{x_n}(t))|) \\ & + |\mathbf{F_r}(q)\left[f(\mathbf{x}(t),\mathbf{u}(t-D)) - f(\mathbf{x_n}(t),\mathbf{u}(t-D))\right]| \\ & + |\mathbf{F_r}(q)\overline{\mathbf{w}}(t)|.\end{aligned}$$

Proof. Rearranging Eq. (6.4),

$$\begin{aligned}\widetilde{\mathbf{w}}(t) &= \widetilde{\mathbf{x}}(t+1) - f(\widetilde{\mathbf{x}}(t),\mathbf{u}(t)) \\ &= \mathbf{x}(t+D+1|t+1) - f(\mathbf{x}(t+D|t),\mathbf{u}(t)),\end{aligned}$$

Given that $\mathbf{x}(t+D|t) = \mathbf{x_n}(t+D)+\mathbf{F_r}(q)(\mathbf{x}(t)-\mathbf{x_n}(t))$ from the NLFSP equations (Eqs. (6.5a) and (6.5b)),

$$\begin{aligned}\widetilde{\mathbf{w}}(t) = & f(\mathbf{x_n}(t+D),\mathbf{u}(t)) + \mathbf{F_r}(q)(\mathbf{x}(t+1)-\mathbf{x_n}(t+1)) \\ & - f(\mathbf{x_n}(t+D) + \mathbf{F_r}(q)(\mathbf{x}(t)-\mathbf{x_n}(t)),\mathbf{u}(t)).\end{aligned}$$

Taking the norm of this last equation, it is possible to write it as

$$\begin{aligned}|\widetilde{\mathbf{w}}(t)| \leq & |f(\mathbf{x_n}(t+D),\mathbf{u}(t)) \\ & - f(\mathbf{x_n}(t+D) + \mathbf{F_r}(q)(\mathbf{x}(t)-\mathbf{x_n}(t)),\mathbf{u}(t))| \\ & + |\mathbf{F_r}(q)(\mathbf{x}(t+1)-\mathbf{x_n}(t+1))|.\end{aligned}$$

Using Assumption 6.1 on this last equation,

$$\begin{aligned}|\widetilde{\mathbf{w}}(t)| \leq & \sigma_x\left(|\mathbf{x_n}(t+D) - \mathbf{x_n}(t+D) + \mathbf{F_r}(q)(\mathbf{x}(t)-\mathbf{x_n}(t))|\right) \\ & + |\mathbf{F_r}(q)(\mathbf{x}(t+1)-\mathbf{x_n}(t+1))| \\ |\widetilde{\mathbf{w}}(t)| \leq & \sigma_x\left(|\mathbf{F_r}(q)(\mathbf{x}(t)-\mathbf{x_n}(t))|\right) \\ & + |\mathbf{F_r}(q)(\mathbf{x}(t+1)-\mathbf{x_n}(t+1))|.\end{aligned}$$

Then, using Eqs. (6.2) and (6.5a),

$$\begin{aligned}|\widetilde{\mathbf{w}}(t)| \leq & \sigma_x \left(|\mathbf{F_r}(q)(\mathbf{x}(t) - \mathbf{x_n}(t))|\right) \\ & + |\mathbf{F_r}(q)[f(\mathbf{x}(t), \mathbf{u}(t-D)) + \overline{\mathbf{w}}(t) - f(\mathbf{x_n}(t), \mathbf{u}(t-D)]| \\ |\widetilde{\mathbf{w}}(t)| \leq & \sigma_x \left(|\mathbf{F_r}(q)(\mathbf{x}(t) - \mathbf{x_n}(t))|\right) \\ & + |\mathbf{F_r}(q)[f(\mathbf{x}(t), \mathbf{u}(t-D)) - f(\mathbf{x_n}(t), \mathbf{u}(t-D)]| \\ & + |\mathbf{F_r}(q)\overline{\mathbf{w}}(t)|,\end{aligned}$$

which completes the proof.

□

A.4 PROOF OF THEOREM 6.2

First, given a signal $\mathbf{a} \in \mathbb{R}_a^n$, the signal sequence is denoted by

$$\mathbf{a}_{[i,j]} \triangleq \{\mathbf{a}(i), \mathbf{a}(i+1), \ldots, \mathbf{a}(j)\},$$

and, with a slight abuse of notation, sometimes $\mathbf{a}$ will also denote a sequence, where the cardinality of the sequence is inferred from the context. $\mathbf{0}_{[i,j]}$ denotes a suitable signal sequence taking a null value.

Using the following common description of a nonlinear system

$$\mathbf{x}(t+1) = f(\mathbf{x}(t), \mathbf{u}(t-d)) + \overline{\mathbf{w}}(t). \tag{A.10}$$

The solution of this system, given an initial state $\mathbf{x}(t)$, a sequence of inputs $\mathbf{u}$ and disturbances $\overline{\mathbf{w}}$ at sampling time $t+j$, is denoted by

$$\mathbf{x}(t+j) = \phi(j, \mathbf{x}(t), \mathbf{u}_{[t-d,t-d+j-1]}, \overline{\mathbf{w}}_{[t,t+j-1]}). \tag{A.11}$$

Note that this solution can be obtained by using (A.10) recursively from time $t+1$ until time $t+j$.

For example,

$$\begin{aligned}\mathbf{x}(t+2) = & f(f(\mathbf{x}(t), \mathbf{u}(t-d)) + \overline{\mathbf{w}}(t), \mathbf{u}(t-d+1)) + \overline{\mathbf{w}}(t+1) \\ = & \phi(2, \mathbf{x}(t), \mathbf{u}_{[t-d,t-d+1]}, \overline{\mathbf{w}}_{[t,t+1]})\end{aligned}$$

Now, the proof of Theorem 6.2.

Theorem. *Consider an open-loop stable nonlinear dead-time system (Assumption 6.2) described by Eq. (6.10) and the dead-time-free model given by Eq. (6.19), where the function $f(.)$ follows Assumption 6.1,*

and there is only one group of states[2]*, i.e.,* $p = 1$*. If the optimal predictor, which utilizes the model described by* (6.20)*, is used, the equivalent disturbance at time* t *is bounded by*

$$|\widetilde{\mathbf{w}}(t)| \leq c_d(|\mathbf{w}(t)|),$$

where c_j *is a recursive* $\mathcal{K}$*-function given by*

$$c_j(|\mathbf{w}(t)|) = |\mathbf{w}(t)| + \sigma_x(|\mathbf{w}(t-1)) + \sigma_x(c_{j-1}(|\mathbf{w}(t)|)), \tag{A.12}$$

with $c_0(|\mathbf{w}(t)|) = |\mathbf{w}(t)|$.

Proof. To obtain the prediction at time $t+1$ it is necessary to multiply Eq. (6.20) by Δ,

$$\begin{aligned}\mathbf{x}(t+1)\Delta =& f(\mathbf{x}(t), \mathbf{u}(t-d))\Delta + \mathbf{e}(t)\\ \mathbf{x}(t+1) - \mathbf{x}(t) =& f(\mathbf{x}(t), \mathbf{u}(t-d))\\ &- f(\mathbf{x}(t-1), \mathbf{u}(t-d-1)) + \mathbf{e}(t).\end{aligned}$$

Given that the expected value of $\mathbf{e}(j)$ for all $j \geq t$ is zero, the optimal prediction is then

$$\mathbf{x}(t+1|t) = f(\mathbf{x}(t|t), \mathbf{u}(t-d)) + \mathbf{x}(t|t) - f(\mathbf{x}(t-1|t), \mathbf{u}(t-d-1)).$$

Note that the expected value of $\mathbf{x}(t)$ at time t, or $\mathbf{x}(t|t)$, is simply the measured states, or $\mathbf{x}(t)$. The same logic can be applied to obtain $\mathbf{x}(t-1|t) = \mathbf{x}(t-1)$. Using these information on the last equation:

$$\begin{aligned}\mathbf{x}(t+1|t) =& f(\mathbf{x}(t), \mathbf{u}(t-d)) + \mathbf{x}(t) - f(\mathbf{x}(t-1), \mathbf{u}(t-d-1))\\ \mathbf{x}(t+1|t) =& f(\mathbf{x}(t), \mathbf{u}(t-d)) + \delta(t),\end{aligned} \tag{A.13}$$

where $\delta(t) = \mathbf{x}(t) - f(\mathbf{x}(t-1), \mathbf{u}(t-d-1))$.

Now, for the prediction at time $t+2$, from the system (6.20),

$$\mathbf{x}(t+2) = f(\mathbf{x}(t+1), \mathbf{u}(t-d+1)) + \frac{\mathbf{e}(t+1)}{\Delta}.$$

[2]The requirement of only one group of states is done to simplify the analysis of the optimal predictor. The results can be extended to the more general case, but the interpretation would be more difficult although the result would be the same, i.e., that the bound on the equivalent disturbance $\widetilde{\mathbf{w}}$ is dependent on the nominal dead time for optimal predictors.

Multiplying again by Δ,

$$\begin{aligned}\mathbf{x}(t+2)\Delta =& f(\mathbf{x}(t+1),\mathbf{u}(t-d+1))\Delta + \mathbf{e}(t+1)\\ \mathbf{x}(t+2)-\mathbf{x}(t+1) =& f(\mathbf{x}(t+1),\mathbf{u}(t-d+1))\\ &- f(\mathbf{x}(t),\mathbf{u}(t-d)) + \mathbf{e}(t+1).\end{aligned}$$

Again, given that the expected value of $\mathbf{e}(j)$ for all $j \geq t$ is zero, the optimal prediction at $t+2$ is then

$$\mathbf{x}(t+2|t) = f(\mathbf{x}(t+1|t),\mathbf{u}(t-d+1)) + \mathbf{x}(t+1|t) - f(\mathbf{x}(t|t),\mathbf{u}(t-d)).$$

Using the same logic applied to obtain $\mathbf{x}(t+1|t)$ and using (A.13) in this last equation

$$\begin{aligned}\mathbf{x}(t+2|t) =& f(\mathbf{x}(t+1|t),\mathbf{u}(t-d+1)) + f(\mathbf{x}(t),\mathbf{u}(t-d)) + \delta(t)\\ &- f(\mathbf{x}(t),\mathbf{u}(t-d))\\ \mathbf{x}(t+2|t) =& f(\mathbf{x}(t+1|t),\mathbf{u}(t-d+1)) + \delta(t).\end{aligned}$$

This prediction can also be expressed using the notation introduced in (A.11):

$$\mathbf{x}(t+2|t) = \phi(2,\mathbf{x}(t),\mathbf{u}_{[t-d,t-d+1]},\delta(t)),$$

with the exception that the disturbance is not a sequence of different values, but fixed ones, i.e., $\boldsymbol{\delta}_{[t,t+j-1]} = \delta(t)$ for all time instants between t and $t+j-1$, and its value is given by A.13.

By inspection, the prediction after the dead-time is then

$$\mathbf{x}(t+d|t) = \phi(d,\mathbf{x}(t),\mathbf{u}_{[t-d,t-1]},\delta(t)). \tag{A.14}$$

As explained in Section 6.3, the equivalent disturbance is given by the auxiliary system Eq. (6.19). But firstly, note that Eq. (6.19a) can be written as

$$\overline{\mathbf{x}}(t+1) = \mathbf{x}_{\mathbf{n}}(t+d+1) = \phi(d+1,\mathbf{x}_{\mathbf{n}}(t),\mathbf{u}_{[t-d,t]},\mathbf{0}). \tag{A.15}$$

Using this notation and substituting (A.14) in Eq. (6.19b),

$$\mathbf{x}(t+d|t) = \phi(d,\mathbf{x}_{\mathbf{n}}(t),\mathbf{u}_{[t-d,t-1]},\mathbf{0}) + \widetilde{\mathbf{w}}(t),$$

then, after applying (A.15) and doing some rearrangements,

$$\widetilde{\mathbf{w}}(t) = \phi(d,\mathbf{x}(t),\mathbf{u}_{[t-d,t-1]},\delta(t)) - \phi(d,\mathbf{x}_{\mathbf{n}}(t),\mathbf{u}_{[t-d,t-1]},\mathbf{0}). \tag{A.16}$$

Hence, the value of $\widetilde{\mathbf{w}}(t)$ is the difference between the evolution of the system considering the measured state at time t as the initial state, while taking into account $\delta(t)$ as the predicted disturbance, and the evolution of the system considering the nominal state without disturbances.

This on itself does not help in the computation of the bound of $\widetilde{\mathbf{w}}$. To obtain this bound, consider the following auxiliary variables

$$\begin{aligned}\mathbf{z}_j &= \phi(j, \mathbf{x}(t), \mathbf{u}_{[t-d,t-d+j-1]}, \delta(t))\\ \overline{\mathbf{z}}_j &= \phi(j, \mathbf{x_n}(t), \mathbf{u}_{[t-d,t-d+j-1]}, \mathbf{0}),\end{aligned}$$

For $j = 1$,

$$\begin{aligned}|\mathbf{z}_1 - \overline{\mathbf{z}}_1| =& |f(\mathbf{x}(t), \mathbf{u}(t-d)) + \mathbf{x}(t) - f(\mathbf{x}(t-1), \mathbf{u}(t-d-1))\\ &- f(\mathbf{x_n}(t), \mathbf{u}(t-d))|,\end{aligned}$$

and given that $\mathbf{x}(t) = \mathbf{x_n}(t) + \mathbf{w}(t)$ from Eq. (6.19b), and that $\mathbf{x_n}(t) = f(\mathbf{x_n}(t-1), \mathbf{u}(t-d-1))$,

$$\begin{aligned}|\mathbf{z}_1 - \overline{\mathbf{z}}_1| \leq\ & |f(\mathbf{x}(t), \mathbf{u}(t-d)) - f(\mathbf{x_n}(t), \mathbf{u}(t-d))|\\ &+ |f(\mathbf{x_n}(t-1), \mathbf{u}(t-d-1)) - f(\mathbf{x}(t-1), \mathbf{u}(t-d-1))|\\ &+ |\mathbf{w}(t)|\end{aligned}$$

Using Assumption 6.1,

$$\begin{aligned}|\mathbf{z}_1 - \overline{\mathbf{z}}_1| \leq& \sigma_x(|\mathbf{x}(t) - \mathbf{x_n}(t)|) + \sigma_x(|\mathbf{x_n}(t-1) - \mathbf{x}(t-1)|) + |\mathbf{w}(t)|\\ \leq& |\mathbf{w}(t)| + \sigma_x(|\mathbf{x_n}(t) + \mathbf{w}(t) - \mathbf{x_n}(t)|)\\ &+ \sigma_x(|\mathbf{x_n}(t-1) - (\mathbf{x_n}(t-1) + \mathbf{w}(t-1))|)\\ \leq& |\mathbf{w}(t)| + \sigma_x(|\mathbf{w}(t)|) + \sigma_x(|\mathbf{w}(t-1)|)\end{aligned}$$

For $j = 2$, using similar procedures as before,

$$\begin{aligned}|\mathbf{z}_2 - \overline{\mathbf{z}}_2| =& |f(\mathbf{z}_1, \mathbf{u}(t-d+1)) + \mathbf{x}(t) - f(\mathbf{x}(t-1), \mathbf{u}(t-d-1))\\ &- f(\overline{\mathbf{z}}_1, \mathbf{u}(t-d+1))|\\ \leq& |f(\mathbf{z}_1, \mathbf{u}(t-d+1)) - f(\overline{\mathbf{z}}_1, \mathbf{u}(t-d+1))| + |\mathbf{w}(t)|\\ &+ |f(\mathbf{x_n}(t-1), \mathbf{u}(t-d-1)) - f(\mathbf{x}(t-1), \mathbf{u}(t-d-1))|\end{aligned}$$

Using, again, Assumption 6.1,

$$|\mathbf{z}_2 - \overline{\mathbf{z}}_2| \leq |\mathbf{w}(t)| + \sigma_x(|\mathbf{w}(t-1)|) + \sigma_x(|\mathbf{z}_1 - \overline{\mathbf{z}}_1|),$$

or, using (A.12),

$$|\mathbf{z}_2 - \overline{\mathbf{z}}_2| \leq c_2(|\mathbf{w}(t)|),$$

By inspection, then,

$$|\mathbf{z}_j - \overline{\mathbf{z}}_j| \leq c_j(|\mathbf{w}(t)|).$$

Applying this last equation on (A.16), after taking its norm,

$$|\widetilde{\mathbf{w}}(t)| = |\mathbf{z}_d - \overline{\mathbf{z}}_d| \leq c_d(|\mathbf{w}(t)|),$$

which completes the proof.

□

A.5 PROOF OF THEOREM 6.3

Theorem. *Consider an open-loop stable nonlinear dead-time system (Assumption 6.2) described by Eq. (6.10) and the dead-time free model given by Eq. (6.19). If the NLFSP structure is used, which is described by Eq. (6.5), the equivalent disturbance is*

$$\widetilde{\mathbf{w}}(t) = \mathbf{F_r}(q)\mathbf{w}(t), \tag{A.17}$$

with $\mathbf{F_r}(q) = F_r(q)\mathbf{I}$*, where* $F_r(q)$ *is a SISO filter.*

Proof. From Eq. (6.5b), which is one of the equations that describe the NLFSP,

$$\mathbf{x}(t+D|t) = \mathbf{x_n}(t+D) + \mathbf{F_r}(q)(\mathbf{x}(t) - \mathbf{x_n}(t)),$$

and, given that, from Eq. (6.10b), $\mathbf{x}(t) = \mathbf{x_n}(t) + \mathbf{w}(t)$,

$$\mathbf{x}(t+D|t) = \mathbf{x_n}(t+D) + \mathbf{F_r}(q)(\mathbf{x_n}(t) + \mathbf{w}(t) - \mathbf{x_n}(t)),$$

then, after making the proper cancellations, the NLFSP equations are reduced to

$$\begin{cases} \mathbf{x_n}(t+D) = f(\mathbf{x_n}(t+D-1), \mathbf{u}(t-1)) & \text{(A.18a)} \\ \mathbf{x}(t+D|t) = \mathbf{x_n}(t+D) + \mathbf{F_r}(q)\mathbf{w}(t) & \text{(A.18b)} \end{cases}.$$

Comparing (A.18) and the dead-time free auxiliary system 6.19

term by term, the following is obtained

$$\widetilde{\mathbf{w}}(t) = \mathbf{F}_{\mathbf{r}}(q)\mathbf{w}(t),$$

which completes the proof. □

B IMPULSE RESPONSE OF FILTERS

Given a filter whose transfer function is represented by $F_r(z)$, by definition, the impulse response can be obtained from the inverse $\mathcal{Z}$-Transform of this filter. Then, the sum of all coefficients β is given by following summation

$$\beta = \sum_{i=0}^{\infty} h_i$$

where h_i is the ith impulse coefficient of the filter.

In order to determine the value of β, the accumulation property of the unilateral $\mathcal{Z}$-Transform can be used [111], which is defined as follows:

$$\mathcal{Z}\left\{\sum_{i=0}^{n} h_i\right\} = \frac{1}{1-z^{-1}} F_r(z).$$

Then, as $F_r(z)$ is a stable filter, in virtue of the final value theorem, the value of β is given by:

$$\beta = \lim_{n\to\infty} \sum_{i=0}^{n} h_i = \lim_{z\to 1} \frac{1-z^{-1}}{1-z^{-1}} F_r(z) = F_r(1) \tag{B.1}$$

This last equation proves that the summation of the coefficients of the impulse response is simply the static gain of the filter $F_r(1)$. Given that throughout this text it was considered that $F_r(1) = 1$, this implies that $\beta = 1$.

Note however, that in Eq. (5.15) the sum of the norm of the coefficients was used, and this sum does not always have the same value as the one obtained through Eq. (B.1).

The equality

$$\sum_{i=0}^{\infty} h_i = \sum_{i=0}^{\infty} |h_i|,$$

will only be satisfied if the sign of the values h_i does not change, which happens when the filter has monotonic behaviour, i.e., it has real poles and does not have dominant zeros. The proof for the general case is beyond the scope of this work, but more information can be found in [112]. However, consider a first order filter with unitary gain given by

$$F_r(z) = \frac{(1-a)}{(1-b)} \frac{(z-b)}{(z-a)}.$$

Its impulse response in the discrete time domain is

$$f_r(t) = \frac{(1-a)}{(1-b)} \left\{ \frac{b}{a}\delta(t) + \left(1 - \frac{b}{a}\right) a^t \mathbb{1}(t) \right\},$$

where $\delta(0) = 1$, $\delta(t) = 0$, for all $t \neq 0$, $\mathbb{1}(t) = 0$, for $t < 0$ and $\mathbb{1}(t) = 1$ for $k \geq 0$. Considering that $a, b > 0$, $|a| < 1$ and $|b| < 1$, if $b < a$, the terms inside the brackets always result in a positive number. If, however, $b > a$, the first coefficient (t=0) is positive and equal to $\frac{(1-a)}{(1-b)}$, but for $t > 0$ the first term inside the brackets is always 0, and $1 - \frac{b}{a}$, which is present in the second term, is negative, resulting in negative coefficients. This will make the sum of the norm of the coefficients greater than their sum.

C INPUT-TO-STATE STABILITY

The ISS property concerns with the continuity of state trajectories on the initial states and the inputs. Roughly speaking, a system is ISS if every state trajectory corresponding to a bounded control remains bounded, and the trajectory eventually becomes small if the input signal is small no matter what the initial state is. The ISS property turns out to be a very natural stability property and, indeed, has been successfully employed in the stability analysis and control synthesis of nonlinear systems with complex structure [99]. The notion of ISS for continuous nonlinear systems was first introduced in [113, 114], and then extended for discrete systems in [99]. Since then, it has been used as a framework for the analysis of the stabilizing properties of MPC, as described in [100].

Before the ISS property can be discussed, consider that $\mathbf{a}_{[i,j]}$ is a signal sequence as defined in Appendix A.4, and, with a slight abuse of notations, sometimes $\mathbf{a}$ will also denote a sequence (the cardinality of the sequence is inferred from the context). For a given sequence

$$||\mathbf{w}|| \triangleq \sup_{t\geq 0} |\mathbf{w}(t)|.$$

Also, the following definition is necessary

Definition C.1. *A function $\beta : \mathbb{R}_{\geq 0} \times \mathbb{Z}_{\geq 0} \to \mathbb{R}_{\geq 0}$, where $\mathbb{Z}_{\geq 0}$ denotes the non-negative integers set, is of class $\mathcal{KL}$ if*

1. *for each fixed $t \geq 0$, $\beta(\cdot, t)$ is of class $\mathcal{K}$;*
2. *for each fixed $s \geq 0$, $\beta(s, \cdot)$ is decreasing and $\beta(s, t) \to 0$ as $t \to +\infty$.*

C.1 PROBLEM STATEMENT

Consider that the plant to be controlled is modeled by a discrete-time invariant nonlinear difference equation as follows

$$\mathbf{x}(t+1) = f(\mathbf{x}(t), \mathbf{u}(t)) + \overline{\mathbf{w}}(t) \tag{C.1}$$

where $\mathbf{x}(t) \in \mathbb{R}^n$ is the system state, $\mathbf{u}(t) \in \mathbb{R}^m$ is the input vector, $\overline{\mathbf{w}}(t) \in \mathbb{R}^n$ is a signal which models external disturbances and mismatches between the real plant and the model, thus, $\overline{\mathbf{w}}(t)$ can be dependent

on the state and input. The solution of the system given by Eq. (C.1) at sampling time t for the initial state $\mathbf{x}(0)$, a sequence of control inputs $\mathbf{u}$ and disturbance $\mathbf{w}$ is denoted as

$$\phi(t, \mathbf{x}(0), \mathbf{u}, \overline{\mathbf{w}}),$$

where $\phi(0, x(0), \mathbf{u}, \mathbf{w}) = \mathbf{x}(0)$ (as was already defined in Eq. (A.11)).

Now, consider that the system is controlled by a certain control law (not necessarily linear) $\mathbf{u}(t) = \kappa(\mathbf{x}(t))$, then the closed-loop system can be expressed as follows:

$$\mathbf{x}(t+1) = f_\kappa(\mathbf{x}(t), \overline{\mathbf{w}}(t)), \tag{C.2}$$

where $f_\kappa(\mathbf{x}(t), \mathbf{w}(t)) \triangleq f(\mathbf{x}(t), \kappa(\mathbf{x}(t))) + \overline{\mathbf{w}}(t)$.

Consider also that the closed-loop system has an equilibrium point at the origin, i.e., $f_\kappa(\mathbf{0}, \mathbf{0}) = \mathbf{0}$.

Then, the ISS property is defined as follows [100]:

Definition C.2. *The system given by Eq. (C.2) is ISS if there exist a $\mathcal{KL}$-function β and a $\mathcal{K}$-function α such that for all initial state $\mathbf{x}(0)$ and sequence of disturbances $\overline{\mathbf{w}} \in \overline{\mathbb{W}}$,*

$$|\phi_\kappa(t, \mathbf{x}(0), \overline{\mathbf{w}})| \leq \beta(|\mathbf{x}(0)|, t) + \alpha(||\overline{\mathbf{w}}_{[0,t-1]}||).$$

The definition of ISS of a system comprises nominal stability and uniformly bounded effect of the uncertainties in a single condition. To check the nominal stability, take $\overline{\mathbf{w}} = \mathbf{0}$, hence the ISS property is reduced to

$$|\phi_\kappa(t, \mathbf{x}(0), \mathbf{0})| \leq \beta(|\mathbf{x}(0)|, t).$$

Since $\beta(a, t)$ is a $\mathcal{KL}$-function, for a fixed a, $\beta(a, t) \to 0$ as $t \to \infty$, which means that the magnitude of the states of the nominal system decreases with t, hence the system is asymptotically stable.

On the other hand, the effect of the uncertainty $\overline{\mathbf{w}}$ makes the system evolution differs from expected. Then it would be desirable that this effect should be bounded and dependent on the size of the uncertainty, which is expressed in the ISS condition via the function α [99]. Note that if the disturbance signal fades, then the disturbed system asymptotically converges to the origin.

Therefore, the ISS notion generalizes existing classic notions on stability of disturbed system allowing the study of the effect of state dependent, persistent or fading disturbances in a single framework [100]. The ISS property is usually demonstrated by means of the existence of

a (not necessarily continuous) Lyapunov function [100].

In some cases, the stability/robustness of the closed-loop system can only be ensured in a neighborhood of the origin and/or for small enough uncertainties. This problem can also be analyzed within the ISS framework by means of the local ISS notion [100].

Definition C.3. *The system given by Eq. (C.2) is said to be locally ISS if there exist constants γ_x and γ, a $\mathcal{KL}$-function β and a $\mathcal{K}$-function α such that*

$$|\phi_\kappa(t, \mathbf{x}(0), \overline{\mathbf{w}})| \leq \beta(|\mathbf{x}(0)|, t) + \alpha(||\overline{\mathbf{w}}_{[0,t-1]}||)$$

for all initial state $|\mathbf{x}(0)| \leq \gamma_x$ and disturbance $|\overline{\mathbf{w}}(t)| \leq \gamma$.

This definition states the existence of a (sufficiently small) state-dependent signal for which the stability of the uncertain system is maintained. Note that this is the nonlinear equivalent of the robustness condition presented in section 2.4.2. In the linear case, the model uncertainties are limited and given by $\Delta P(z)$, in the nonlinear case, the disturbance is limited by γ.

Throughout this thesis, it will be of no concern the proof that a given closed-loop system is ISS, but rather, it will be assumed that the controller was designed in a way that it is ISS, or, at least, locally ISS with constants γ_x and γ that are not necessarily known.

C.2 ISS FOR THE ALTERNATIVE SYSTEM DESCRIPTION

Considering the system described by

$$\begin{cases} \mathbf{x_n}(t+1) = f(\mathbf{x_n}(t), \mathbf{u}(t-D)) & \text{(C.3a)} \\ \mathbf{x}(t) = \mathbf{x_n}(t) + \mathbf{w}(t), & \text{(C.3b)} \end{cases}$$

note that, if $\mathbf{w}$ is bounded, the only way that $\mathbf{x}$ can diverge (instability) is if $\mathbf{x_n}$ diverges. Given that the control law is $\mathbf{u}(t) = \kappa(\mathbf{x}(t)) = \kappa(\mathbf{x_n}(t) + \mathbf{w}(t))$, Eq. (C.3a) can be written as

$$\mathbf{x_n}(t+1) = f(\mathbf{x_n}(t), q^{-d_n}\kappa(\mathbf{x_n}(t) + \mathbf{w}(t))) = f_\kappa(\mathbf{x_n}(t), \mathbf{w}(t)).$$

Defining the evolution of this system as

$$\phi_\kappa(t, \mathbf{x_n}(0), \mathbf{w}),$$

then the system given by Eq. (C.3), with control law $\kappa(\mathbf{x}(t))$ is ISS if there exist a $\mathcal{KL}$-function β and a $\mathcal{K}$-function α such that for all initial state $\mathbf{x_n}(0) = \mathbf{x}(0)$ and sequence of disturbances $\mathbf{w} \in \mathbb{W}$,

$$|\phi_\kappa(t, \mathbf{x}(0), \mathbf{w})| \leq \beta(|\mathbf{x}(0)|, t) + \alpha(||\mathbf{w}_{[0,t-1]}||),$$

as of Definition C.2.

It is also interesting to prove that, given a bounded input disturbance, i.e., $|\overline{\mathbf{w}}| \leq \gamma$, the alternative output disturbance $\mathbf{w}$ is also bounded. To do this, equating Eq. (C.1) and Eq. (C.3),

$$\begin{aligned} f(\mathbf{x}(t-1), \mathbf{u}(t-1)) + \overline{\mathbf{w}}(t-1) &= \mathbf{x_n}(t) + \mathbf{w}(t) \\ f(\mathbf{x}(t-1), \mathbf{u}(t-1)) + \overline{\mathbf{w}}(t-1) &= f(\mathbf{x_n}(t-1), \mathbf{u}(t-1)) + \mathbf{w}(t) \end{aligned}$$

$$\mathbf{w}(t) = f(\mathbf{x}(t-1), \mathbf{u}(t-1)) - f(\mathbf{x_n}(t-1), \mathbf{u}(t-1)) + \overline{\mathbf{w}}(t-1). \tag{C.4}$$

Considering that the system function $f(.)$ follows Assumption 6.1, i.e.,

$$|f(\mathbf{x_1}, \mathbf{u}) - f(\mathbf{x_2}, \mathbf{u})| \leq \sigma_x(|\mathbf{x_1} - \mathbf{x_2}|),$$

and taking the norm of Eq. (C.4),

$$\begin{aligned} |\mathbf{w}(t)| &\leq |f(\mathbf{x}(t-1), \mathbf{u}(t-1)) - f(\mathbf{x_n}(t-1), \mathbf{u}(t-1))| \\ &\quad + |\overline{\mathbf{w}}(t-1)| \\ |\mathbf{w}| &\leq \sigma_x(|\mathbf{x} - \mathbf{x_n}|) + |\overline{\mathbf{w}}| \\ |\mathbf{w}| &\leq \sigma_x(|\mathbf{x} - \mathbf{x_n}|) + \gamma. \end{aligned}$$

Given that, in practice, the states are also bounded in a set $\mathbb{X}$, i.e., $\mathbb{X} = \{\mathbf{x} \in \Re^n : |\mathbf{x}| \leq \gamma_x\}$, then $\sigma_x(|\mathbf{x} - \mathbf{x_n}|) \leq \overline{\gamma}$. Hence, $\mathbf{w}$ is also bounded:

$$|\mathbf{w}| \leq \overline{\gamma} + \gamma.$$